

UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE QUITO

CARRERA: INGENIERÍA DE SISTEMAS

Trabajo de titulación previo a la obtención del título de:
INGENIERA DE SISTEMAS E INGENIERO DE SISTEMAS

TEMA:

**ANÁLISIS, DISEÑO, CONSTRUCCIÓN Y EVALUACIÓN DE UN
SISTEMA EXPERTO TRADUCTOR DE LOS IDIOMAS:
CASTELLANO A KICHWA ECUATORIANO BASADO EN REGLAS
DE INFERENCIA Y RAZONAMIENTO APROXIMADO.**

AUTORES:

PAMELA CECILIA GÓMEZ MARTÍNEZ
PAÚL ALEXANDER FREIRE JARAMILLO

DIRECTOR:

GONZALO BAYARDO CAMPUZANO NIETO

Quito, octubre de 2014

DECLARATORIA DE RESPONSABILIDAD Y AUTORIZACIÓN DE USO DEL TRABAJO DE TITULACIÓN

Nosotros; autorizamos a la Universidad Politécnica Salesiana la publicación total o parcial de este trabajo de titulación y su reproducción sin fines de lucro.

Además, declaramos que los conceptos, análisis desarrollados y las conclusiones del presente trabajo son de exclusiva responsabilidad los autores.

Quito, octubre de 2014

Pamela Cecilia Gómez Martínez

C.C. 1713140679

Paúl Alexander Freire Jaramillo

C.C. 1717255754

DEDICATORIA

A Dios por dotarnos de salud y sabiduría en la realización de este trabajo.

Con todo mi corazón quiero dedicar este trabajo a mi hija Paula Nicole, a mi madre, a mi padre por estar pendiente de todo lo que necesito, a mi hermana por ser la única persona en apoyarme en el cuidado diario de mi hija, sin ella no lo hubiera logrado, a mi esposo Paúl por ser un compañero inmejorable, estar en la buenas y en las malas aparte de que compartimos esta grandiosa profesión, estamos juntos en este trabajo de fin de carrera, no puedo dejar a un lado mis hermanos que ellos me han colaborado con mi chiquita, gracias a todos por su incansable cariño.

Pamela Gómez Martínez

A mi compañera y esposa Pamela, que con su inmenso e incondicional cariño ha estado en los momentos buenos y malos de mi vida, ahora estamos juntos en la culminación de este proyecto, a mi hija Nicole, por brindarme gran parte de su tiempo, permitiendo que realice este trabajo, y en especial a todas las personas que creen en mis capacidades para culminar mi carrera, entre ellos mis padres.

Paúl Freire Jaramillo

AGRADECIMIENTO

Este proyecto es el resultado de un trabajo conjunto, por esta razón agradecemos al Físico Bayardo Campuzano, que en calidad de director de trabajo de titulación, con sus conocimientos y gran paciencia ha sabido guiarnos en el desarrollo del presente trabajo.

A la Ingeniera Patricia Zapata, por su incondicional apoyo moral e intelectual en el diseño del proyecto.

Al Doctor Rommel Sánchez, experto en el idioma Kichwa por su gran disponibilidad y paciencia en la enseñanza de dicho idioma.

Pamela Gómez Martínez

Paúl Freire Jaramillo

ÍNDICE

INTRODUCCIÓN.....	1
CAPÍTULO 1.....	5
MARCO TEÓRICO	5
1.1. Gramática	5
1.1.1. AlliKillkay– ortografía.....	5
1.1.2. Kichwarimakillkay– Gramática Kichwa.....	6
1.1.2.1. Características de las lenguas.....	6
1.1.3. Las partes de la oración.....	7
1.1.3.1. El artículo.....	8
1.1.3.2. El adjetivo.	8
1.1.3.3. El verbo.	8
1.1.3.4. El nombre o sustantivo.....	9
1.1.3.5. El plural.....	9
1.1.3.6. Los pronombres personales.....	10
1.1.3.7. Los pronombres demostrativos.	10
1.1.3.8. Los pronombres posesivos.	11
1.1.4. La palabra.....	11
1.1.4.1. Clases de palabras en Castellano y Kichwa.	12
1.1.4.2. Lexema o raíz.....	12
1.1.4.3. Morfema o afijo.....	12
1.1.4.3.1.Prefijos.	13
1.1.4.3.2.Sufijos.	13
1.1.4.3.3.Infijos.	14
1.1.5. Conjugaciones.....	14
1.1.5.1. Terminaciones verbales para los tres tiempos.....	15
1.1.5.2. Presente simple.....	15
1.1.5.3. Pasado perfecto simple.....	16
1.1.5.4. Futuro.	16
1.1.6. La importancia del verbo kana.	17
1.1.7. Chu (Interrogativo).....	17
1.1.8. Como responder.	18

1.1.9. Sufijo pish, pash (y, también).	19
1.1.10. Sufijo manta (de, desde).	20
1.1.11. Sufijo kama (hasta).	20
1.1.12. Sufijo intin (junto con).	21
1.1.13. Sufijo man (a, hacia).	21
1.1.14. Sufijo wan (con).	21
1.2. Introducción a la Inteligencia Artificial	22
1.2.1. Principales áreas de la inteligencia artificial.	22
1.3. Introducción al procesamiento de lenguaje natural	22
1.3.1. Historia.	23
1.3.2. Análisis.	23
1.3.2.1. Morfología.	23
1.3.2.2. Sintaxis.	23
1.3.2.3. Semántica.	24
1.3.2.4. Pragmática.	24
1.3.3. Arquitectura.	25
1.3.4. Principales dificultades del procesamiento del lenguaje natural.	26
1.3.4.1. Ambigüedades.	26
1.3.4.2. Detección de espacios entre palabras.	26
1.3.4.3. Recepción imperfecta de datos.	27
1.3.5. Aplicaciones del procesamiento de lenguaje natural.	27
1.3.5.1. Traducción automática.	27
1.3.5.1.1. Historia.	27
1.3.5.1.2. Análisis.	28
• Traducción basada en reglas	29
• Traducción basada en estadística	29
1.3.6. Modelamiento del procesamiento del lenguaje natural.	31
1.3.6.1. Modelamiento de los idiomas Castellano y Kichwa	31
1.4. Introducción a los sistemas expertos	37
1.4.1. Historia.	37
1.4.2. Definiciones de sistemas expertos.	37
1.4.3. Estructura básica de un sistema experto.	38
1.4.4. Arquitectura de un sistema experto.	39

1.4.5.	Ventajas y limitaciones de los sistemas expertos.....	39
1.4.5.1.	Ventajas.....	39
1.4.5.2.	Limitaciones.....	40
1.4.6.	Tipos de sistemas expertos.....	41
1.4.6.1.	Sistemas expertos basados en reglas.....	41
1.4.6.1.1.	Antecedentes.....	41
1.4.6.1.2.	Definición de reglas.....	42
1.4.6.1.3.	Encadenamiento hacia adelante.....	42
1.4.6.1.4.	Encadenamiento hacia atrás.....	43
1.4.6.2.	Sistemas expertos basados en probabilidad.....	43
1.4.6.2.1.	Antecedentes.....	43
1.4.6.2.2.	Sistemas expertos basados en razonamiento aproximado.....	44
•	Método de probabilidad clásico.....	45
•	Redes bayesianas.....	45
•	Modelo de factores de certeza MYCIN.....	46
•	Lógica difusa.....	46
CAPÍTULO 2	47
ANÁLISIS, DISEÑO Y CONSTRUCCIÓN DEL SISTEMA	47
2.1.	Introducción.....	47
2.2.	Análisis.....	47
2.2.1.	Análisis de viabilidad.....	47
2.2.1.1.	Viabilidad técnica.....	48
2.2.1.2.	Viabilidad económica.....	48
2.2.2.	Análisis de requerimientos.....	49
2.2.2.1.	Prospectiva del sistema.....	49
2.2.2.2.	Funciones del sistema.....	49
2.2.2.3.	Restricciones.....	51
2.2.3.	Requerimientos funcionales.....	51
2.2.3.1.	Usuario del sistema.....	51
Requerimiento funcional 1.1.	Interfaz perfil de usuario.....	51
Requerimiento funcional 1.2.	Traducción coherente.....	52
Requerimiento funcional 1.3.	Traducción con detección de ambigüedades léxicas..	52
Requerimiento funcional 1.4.	Traducción sugerida.....	53

Requerimiento funcional 1.5. Traducción en presente, pasado y futuro.....	53
2.3. Diseño	54
2.3.1. Arquitectura del sistema experto.....	54
2.3.2. Estructura del sistema.	55
2.3.2.1. Actores.	55
2.3.2.1.1.Experto.	56
2.3.2.1.2.Ingeniero del conocimiento.....	56
2.3.2.1.3.Usuario.	56
2.3.2.2. Conocimiento del sistema.	57
2.3.2.2.1.Adquisición del conocimiento.....	57
• Etapas de la adquisición del conocimiento	57
2.3.2.2.2.Base del conocimiento.	58
2.3.2.2.3.Base de hechos.	60
2.3.2.2.4.Motor de inferencia.....	61
• Algoritmo del motor de inferencia.....	62
2.3.2.3. Interfaz de usuario.....	63
2.3.2.3.1.Diseño de la interfaz de usuario.	63
2.3.2.4. Módulo de justificación.....	65
2.3.2.5. Cliente web.....	65
2.3.2.6. Base de datos.....	66
2.3.2.7. Mapa de navegación.....	66
2.3.2.7.1.Mapa de navegación para traducción.	66
2.3.3. Diagramas de casos de uso.....	67
2.3.3.1. Diagrama de casos de uso experto.	67
2.3.3.2. Diagrama de casos de uso del ingeniero del conocimiento.....	68
2.3.3.3. Diagrama de casos de uso del usuario.....	69
2.3.4. Diagrama de flujo del sistema.....	70
2.3.5. Diagrama de clases del sistema.....	70
2.4. Construcción	71
2.4.1. Clase conexiones.....	71
2.4.2. Clase funciones.	75
2.4.3. Clase: default.....	79
2.4.4. Clase reglas de inferencia (sistema experto).....	81

2.4.5.	Clase ambigüedades.....	84
2.4.6.	Clase módulo de justificación (sistema experto).	85
CAPÍTULO 3.....		86
PRUEBAS, RESULTADOS Y EVALUACIÓN.....		86
3.1.	Pruebas	86
3.1.1.	Pruebas contra requerimientos.	86
3.1.1.1.	Requerimientos funcionales.....	86
3.2.	Resultados	88
3.2.1.	Acceso al sistema.	88
3.2.1.1.	Módulo 1.....	89
3.2.1.2.	Módulo 2.	90
3.3.	Evaluación.....	96
3.2.1.	Modelo de la evaluación.	96
3.2.2.	Resultados de la evaluación.	97
CONCLUSIONES.....		99
RECOMENDACIONES.....		101
LISTA DE REFERENCIAS		102

ÍNDICE DE TABLAS

Tabla 1. Clases de palabras del Castellano y el Kichwa	12
Tabla 2. Terminaciones verbales del idioma Kichwa según el pronombre	15
Tabla 3. Traducción automática basada en reglas frente a la estadística	30
Tabla 4. Detalle especificaciones de hardware	48
Tabla 5. Detalle viabilidad económica.....	49
Tabla 6. Requerimiento funcional: Interfaz perfil de usuario	51
Tabla 7. Requerimiento funcional: Traducción coherente	52
Tabla 8. Requerimiento funcional: Traducción con detección de ambigüedades léxicas.....	52
Tabla 9. Requerimiento funcional: Traducción mejorada.....	53
Tabla 10. Requerimiento funcional: Traducción en presente, pasado y futuro	53
Tabla 11. Pruebas contra requerimientos	86

ÍNDICE DE FIGURAS

Figura 1. Análisis sintáctico de la oración	24
Figura 2. Análisis bajo los cuatro parámetros del PLN	25
Figura 3. Modelamiento lenguaje natural la oración Castellano.....	31
Figura 4. Ejemplo modelamiento lenguaje natural la oración Castellano	31
Figura 5. Modelamiento lenguaje natural la oración Kichwa.....	32
Figura 6. Ejemplo Modelamiento lenguaje natural la oración Kichwa	32
Figura 7. Modelamiento lenguaje natural pronombre Castellano.....	33
Figura 8. Modelamiento lenguaje natural pronombre Kichwa	33
Figura 9. Modelamiento lenguaje natural artículo Castellano	34
Figura 10. Modelamiento lenguaje natural artículo Kichwa.....	34
Figura 11. Modelamiento lenguaje natural adjetivo Castellano.....	35
Figura 12. Modelamiento lenguaje natural adjetivo Kichwa	35
Figura 13. Modelamiento lenguaje natural plural Castellano	36
Figura 14. Modelamiento lenguaje natural plural Kichwa.....	36
Figura 15. Arquitectura básica de un sistema experto	39
Figura 16. Etapas para el desarrollo de un sistema experto	40
Figura 17. Ejemplo red bayesiana.....	45
Figura 18. Arquitectura del sistema	55
Figura 19. Diagrama base de hechos.....	60
Figura 20. Algoritmo de motor de inferencia.....	61
Figura 21. Diseño de la interfaz de usuario.....	64
Figura 22. Diseño del módulo de justificación	65
Figura 23. Mapa de navegación usuario	66
Figura 24. Diagrama de casos de uso experto.....	67
Figura 25. Diagrama de casos de uso ingeniero del conocimiento.....	68
Figura 26. Diagrama de casos de uso usuario	69
Figura 27. Diagrama de flujo del sistema	70
Figura 28. Diagrama de clases	70
Figura 29. Clase conexiones	71
Figura 30. Clase variables globales.....	72
Figura 31. Función conectar.....	72
Figura 32. Función traduce verbo	73
Figura 33. Función pronombre.....	73
Figura 34. Función tipo término.....	74
Figura 35. Función traduce	74
Figura 36. Función busca ambigüedad.....	75
Figura 37. Clases funciones	75
Figura 38. Función separa palabras.....	76
Figura 39. Función elimina tildes.....	76
Figura 40. Función limpia variables.....	77
Figura 41. Función comprueba verbo	77

Figura 42. Función singular	78
Figura 43. Función plurales.....	78
Figura 44. Función primera mayúscula.....	78
Figura 45. Función cambio de letras	79
Figura 46. Función conteo.....	79
Figura 47. Clase default	80
Figura 48. Función button1	80
Figura 49. Función button2	81
Figura 50. Clase reglas de inferencia	82
Figura 51. Función aplica reglas	82
Figura 52. Función sufijo verbo	83
Figura 53. Función párrafo orden.....	83
Figura 54. Función números	84
Figura 55. Clase ambigüedades.....	84
Figura 56. Función controla ambigüedades	85
Figura 57. Clase módulo de justificación.....	85
Figura 58. Interfaz de usuario	88
Figura 59. Módulo 1.....	89
Figura 60. Módulo 2.....	90
Figura 61. Módulo 2.....	91
Figura 62. Módulo 2.....	92
Figura 63. Módulo 2.....	92
Figura 64. Módulo 2.....	93
Figura 65. Módulo 2.....	93
Figura 66. Módulo 2.....	94
Figura 67. Módulo 2.....	94

ÍNDICE DE ANEXOS

Anexo 1. Evaluación del traductor Castellano – Kichwa (modelo 1).....	104
Anexo 2. Evaluación del traductor Castellano – Kichwa (modelo 2).....	105
Anexo 3. Evaluación del traductor Castellano – Kichwa (modelo 3).....	106
Anexo 4. Evaluación del traductor Castellano – Kichwa (modelo 4).....	107
Anexo 5. Evaluación del traductor Castellano – Kichwa (modelo 5).....	108

RESUMEN

El producto realizado es un software de traducción de los idiomas Castellano – Kichwa ecuatoriano, dicho software cuenta con una interfaz simple la cual facilita la comunicación entre éste y el usuario, el mismo que simplemente ingresa texto en el idioma Castellano, pulsa el botón traducir, y obtiene la respuesta de traducción al idioma Kichwa ecuatoriano más acertada y entendible posible.

En cuanto a la arquitectura y tipo de software el producto fue construido en base a un sistema experto, el mismo que estará basado en dos métodos como son:

- Reglas de inferencia
- Razonamiento aproximado

Se usa estos métodos ya que en base a la investigación y estudios que se realizaron, son los más acertados para un software de traducción.

Cabe mencionar que primero se construyó un software sistema experto traductor de los idiomas Castellano-Kichwa ecuatoriano basado en reglas de inferencia y tratamos de mejorarlo anexando al sistema el método de razonamiento aproximado, posteriormente se evaluó al sistema por expertos en el idioma Kichwa, mediante lo cual se comprobó su mejora y su efectividad siendo esto nuestro principal objetivo en la elaboración del trabajo de titulación.

Se pretendió encontrar la manera que la traducción sea lo más coherente posible; hacer que el sistema sea amigable para el usuario y plasmar a esta lengua ancestral ecuatoriana en una herramienta actualizada mediante la cual cualquier persona puede comunicarse con un Kichwa hablante ecuatoriano. Así mismo, se pretendió promover el uso de este idioma con la elaboración de esta aplicación.

ABSTRACT

The made product is a translation software of the Castilian to Ecuadorian Kichwa, this software has a simple interface which facilitates communication between it and the user, simply enter the same text in the Castilian language, press the translate button, and gets the reply translation more accurate and understandable language Kichwa Ecuadorian possible.

As for the type of software architecture and the product was built based on an expert system, the same will be based on two methods as:

- Rules of inference
- Approximate Reasoning

These methods are used and that based on research and studies conducted, are the most successful for translation software.

It is noteworthy that the first software translator of the Castilian-Kichwa language Ecuadorian based on inference rules and try to improve it by appending the system method approximate reasoning expert system was built, it was subsequently evaluated the system by experts in the Kichwa language, thereby which improvement and its effectiveness was proven this being our main objective in developing the job qualifications.

It was intended to find ways that the translation is as consistent as possible; make the system user friendly and translate this Ecuadorian ancestral language in an updated tool through which anyone can communicate with a Kichwa Ecuadorian speaker. It also was intended to promote the use of this language to the development of this application.

INTRODUCCIÓN

- **Planteamiento del problema**

Actualmente la tecnología ha intervenido en gran parte en la vida del ser humano, facilitando así la comunicación en varios idiomas mediante traductores de lenguaje, los cuales pueden ser encontrados en gran variedad.

Se ve ahí la necesidad de desarrollar una herramienta tecnológica de esta índole, la cual realice la traducción básica de los idiomas Castellano - Kichwa ecuatoriano, usando un sistema experto basado en Reglas de Inferencia y Razonamiento Aproximado. El objetivo principal es determinar qué calidad de traducción se obtiene utilizando estos métodos en lo que se refiere al procesamiento de lenguaje natural, siendo evaluados de manera cualitativa con expertos en el idioma Kichwa.

Se pretende encontrar la manera que la traducción sea lo más coherente posible; hacer que el sistema sea amigable para el usuario y plasmar a esta lengua ancestral ecuatoriana en una herramienta actualizada mediante la cual cualquier persona podrá comunicarse con un Kichwa hablante ecuatoriano. Así mismo, promover el uso de este idioma.

- **Objetivos**

- **Objetivo general**

Analizar, diseñar, construir y evaluar un sistema experto traductor de los idiomas Castellano - Kichwa ecuatoriano.

- **Objetivos específicos**

- Desarrollar un traductor de los idiomas Castellano - Kichwa ecuatoriano basado en reglas de inferencia y tratar de mejorarlo aplicando razonamiento aproximado, para la web.

- Utilizar la metodología orientada a objetos y UML (lenguaje unificado de modelado) para el análisis y diseño del sistema.
- Desarrollar una base de hechos en MySQL tomando en cuenta los conocimientos del experto humano.
- Realizar un estudio básico de la gramática del idioma Kichwa ecuatoriano.
- Analizar la gramática del lenguaje Castellano.
- Elaborar un sistema que realice una traducción de buena calidad, es decir, entendible para el usuario.
- Evaluar los resultados obtenidos cualitativamente con un experto humano, para determinar la calidad de traducción.
- **Justificación**

El presente proyecto está realizado con la finalidad de realzar la lengua nativa ecuatoriana (Kichwa), para lo cual se realizó un estudio de dicho lenguaje y su cultura, mediante lo cual se llegó a la conclusión que el idioma Kichwa no tiene regularización lingüística ni normativas dirigidas a fomentar la unidad idiomática, ya que las personas Kichwa hablantes tienen diferentes ideologías y no han logrado unificarlas en un solo documento y así garantizar una norma común en el idioma Kichwa, razón por la cual se encontrarán varios textos del idioma Kichwa ecuatoriano en donde se podrá divisar varias diferencias en la forma de hablar y escribir dicho idioma.

Este proyecto recopila las reglas de los textos de Kichwa ecuatoriano más generales posibles para la traducción Kichwa – Castellano y pretende realizar un trabajo acertado, apegándose lo más posible a la ideología y cultura indígena ecuatoriana, razón por la cual la traducción será genérica acorde con los textos antes mencionados.

- **Alcance del proyecto**

El producto que se va a realizar es un software de traducción de los idiomas Castellano–Kichwa ecuatoriano, dicho software contará con una interfaz desarrollada en ambiente web, la cual facilitará la comunicación entre éste y el usuario, el mismo que simplemente ingresará texto en el idioma Castellano, pulsará el botón traducir y obtendrá la respuesta de traducción al idioma Kichwa ecuatoriano más acertada y entendible posible.

En cuanto a la arquitectura y tipo de software el producto va a ser construido en base a un sistema experto, el mismo que estará basado en dos métodos como son:

- Reglas de inferencia
- Razonamiento aproximado

Se va a usar estos métodos ya que en base a la investigación y estudios que se realizaron son las más acertadas para un software de traducción.

Cabe mencionar que primero se construirá un software sistema experto traductor de los idiomas Castellano - Kichwa ecuatoriano basado en reglas de inferencia y se tratará de mejorarlo anexando al sistema el método de razonamiento aproximado, posteriormente se evaluará al sistema por expertos en el idioma Kichwa, mediante lo cual se podrá comprobar su mejora y su efectividad siendo esto nuestro principal objetivo en la elaboración del trabajo de titulación.

- **Metodología**

La programación orientada a objetos es un paradigma de programación que usa objetos y sus interacciones para diseñar aplicaciones y programas de computadora.

Está basado en varias técnicas, incluyendo herencia, modularidad, polimorfismo y encapsulamiento. Su uso se popularizó a principios de la década de 1990. Actualmente son muchos los lenguajes de programación que soportan la orientación a objetos.

Esto difiere de la programación estructurada tradicional, en la que los datos y los procedimientos están separados y sin relación, ya que lo único que se busca es el procesamiento de unos datos de entrada para obtener otros de salida. La programación estructurada anima al programador a pensar sobre todo en términos de procedimientos o funciones, y en segundo lugar en las estructuras de datos que esos procedimientos manejan. En la programación estructurada sólo se escriben funciones que procesan datos. Los programadores que emplean éste nuevo paradigma, en cambio, primero definen objetos para luego enviarles mensajes solicitándoles que realicen sus métodos por sí mismos.

- **Lenguaje Modelado Unificado (UML).**

Una exigencia de la gran mayoría de software trabaja bajo una arquitectura en capas, para que pueda formalizarse con un lenguaje estándar y unificado.

Es decir, se requiere que cada una de las partes que comprende el desarrollo de todo software de diseño orientado a objetos, se visualice, especifique y documente con lenguaje común.

Se necesitaba un lenguaje que fuese gráfico, a fin de especificar y documentar un sistema de software, de un modo estándar incluyendo aspectos conceptuales tales como procesos de negocios y funciones del sistema.

Este lenguaje unificado que cumple con estos requerimientos, es ciertamente UML, el cual cuenta con una notación estándar y semánticas esenciales para el modelado de un sistema orientado a objetos.

“UML es un conjunto de herramientas, que permite modelar (analizar y diseñar) sistemas orientados a objetos”(mafu, 2013).

CAPÍTULO 1

MARCO TEÓRICO

1.1. Gramática

1.1.1. AlliKillkay– ortografía.

- **Killkayllika – alfabeto gráfico**

El alfabeto kichwa está formado por 20 grafías aprobadas por la Dirección de Educación Intercultural Bilingüe DINEIB que consta de 3 vocales y 17 consonantes.

Uyaywakuna - Vocales a, i, u

Uyantikuna - Consonantes: ch, h, k, l, ll, m, n, ñ, p, r, s, sh, t, ts, w, y, z.

Cabe mencionar que se necesario realizar un estudio más avanzado del idioma, ya que no se ha podido llegar a un acuerdo social entre los hablantes de los diferentes dialectos Kichwas del Ecuador; tomando en cuenta que en estos fonemas aprobados por el DINEIB no se encuentran todos los sonidos del alfabeto fonológico necesario del idioma Kichwa ecuatoriano para que de tal manera los pueblos y naciones andinas puedan expresar correctamente su pensamiento; ya que la intelectualidad indígena no pone todo su empeño en construir un sistema de educación propio de los pueblos indígenas, al idioma Kichwa solo han convertido en una herramienta de transmisión de otros conocimientos de fuera que no responden a las necesidades de sus pueblos.

En el trabajo investigativo se utilizará algunos conceptos lingüísticos del Castellano ya que algunos términos Kichwas merecen un análisis más detallado ya que son: una traducción directamente del Castellano, otras son creaciones forzosas.

1.1.2. Kichwarimakillkay– Gramática Kichwa.

1.1.2.1. Características de las lenguas.

Se tratará a la gramática castellana y a la kichwa a la par para determinar semejanzas y diferencias.

Existen categorías propias de cada idioma, pero alguna de estas es común que se repitan en otros idiomas. Así se tiene las variantes entre la estructura gramatical de la lengua kichwa y la castellana.

Estructura de una oración de la lengua kichwa tiene el siguiente orden:

Primero "S" sujeto, segundo "C" complemento, tercero "V" verbo.

Mientras que la estructura de la lengua castellana marca el siguiente orden:

Primero "S" sujeto, segundo "V" verbo, tercero "C" complemento.

Ejemplos

Kichwa

Pedroka tantata mikun.

S C V

Castellano

Pedro come pan

S V C

Sujeto: en ambas lenguas tienen la misma función, es decir es del nombre o sustantivo, de quién se habla en la oración.

Complemento: en ambas lenguas tienen la misma función, es decir es el que interactúa con las diferentes partes de la oración, para que complemente su sentido.

Los complementos se construyen de una o más palabras, adjetivos, adverbios, verbos auxiliares, etc.

Verbo: en ambas lenguas tiene la misma función, es decir es la parte de la oración que indica acción

El Kichwa es una lengua compleja ya que agrupan en una sola palabra varios elementos por ejemplo a la raíz de la palabra se le añaden sufijo, infijos, prefijos, es por ello que no se puede hacer traducciones exactas al Castellano ya que hay palabras que no se pueden remplazar del Castellano al Kichwa y viceversa por ser lenguas construidas desde pensamientos distintos.

“El pensamiento andino es más colectivo, comunitario, complementario, simbólico y metafórico”.(Paza, 2011, pág. 11)

Escritura de nombres propios.

Los nombres propios reciben un trato especial para la traducción; se los puede conservar como esta en el Castellano o el nombre sufre una variación importante, como en la lengua Kichwa existe solo tres vocales a, i, o, se cambian la e→i, y la o→u.

Kichwa	Castellano
Rummil	Rommel
Mercedes	Micidis

1.1.3. Las partes de la oración.

En la lengua Kichwa la oración tiene 3 componentes: sujeto, complemento, verbo (S.C.V.) dentro de ello se encuentra diferentes clases de palabras que con la aglutinación de los nombres y morfemas podemos ir construyendo la oración, como:

el artículo, sustantivo, pronombre, verbo, adjetivo, adverbio, morfemas, posposición, conjunción, participio, e interjección, entre otras.

1.1.3.1. El artículo.

El artículo acompaña al sustantivo; en el Castellano hace las veces de: el, la, los, las; mientras que en el Kichwa se convierte en un sufijo **ka** que va postpuesta al sustantivo.

Kichwa

misika

Sisaka

Castellano

El gato

La flor

1.1.3.2. El adjetivo.

Es la palabra que califica al sustantivo; en el Castellano va después del sustantivo mientras que en el Kichwa va antes del sustantivo.

Kichwa

Puka wasi

Kuyaylla sisa

Castellano

Casa roja

Flor bonita

1.1.3.3. El verbo.

El verbo es la parte de la oración que expresa existencia, acción, consecución, condición o estado del sujeto, semánticamente expresa una predicación completa.

En el idioma Kichwa la acción que se realiza se caracteriza mediante la terminación –na, en el castellano son los infinitivos -ar, -er, -ir.

Kichwa	Castellano
Kuyana	amar
Chapana	mirar
Kana	ser o estar
Kawsana	vivir
Mikuna	comer

1.1.3.4. El nombre o sustantivo.

Palabra que sirve para designar personas, animales, o cosas; Puede funcionar como sujeto de la oración.

En el idioma Kichwa el sustantivo toma las mismas funciones que en el Castellano simplemente tiene una traducción directa.

Kichwa	Castellano
Wasi	Casa
Inti	Sol
Allku	Perro

1.1.3.5. El plural.

El plural en el Castellano se caracteriza por ser el sustantivo que termina en **s**; mientras que para pluralizar los sustantivos en el idioma Kichwa se utiliza el morfema "**-kuna**".

Singular	Plural
Wasi – casa	Was kuna – casas
Yura – árbol	Yur kuna – árboles

1.1.3.6. Los pronombres personales.

Los pronombres personales indican o hacen referencia a las personas o sujetos; en el idioma Kichwa los pronombres personales son los siguientes:

Singular

Kichwa	Castellano
Ñukaka	yo
Kikinka	usted (por respeto)
Kanka	tú (por confianza)
Payka	él/ella

Plural

Kichwa	Castellano
Ñukanchika	nosotros, nosotras
Kankunaka	ustedes
Kikinkunaka	vosotros, ustedes
Paykunaka	ellos/ellas

1.1.3.7. Los pronombres demostrativos.

Los pronombres demostrativos sirven para hacer referencia a las personas, animales o cosas; en el idioma Kichwa los pronombres demostrativos son los siguientes:

Kichwa	Castellano
Kay	Este, esta, esto.
Chay	Ese, esa, eso, aquel, aquello/lla.
Kayshuk	Este otro, esta otra.
Chayshuk	Ese otro, esa otra, aquel otro.

Kaynik	El de más acá, la de más acá.
Chaynik	El de más allá, la de más allá.

Ejemplos

Kichwa	Castellano
Kay wasipi kawsani	Vivo en esta casa.
Chay wasipi kawsani	Vivo en esa casa.
Esa flor es roja	Chay sisa pucca kan

1.1.3.8. Los pronombres posesivos.

Los pronombres posesivos, indican posesión o pertenencia. En el idioma Kichwa los pronombres posesivos son los siguientes:

Kichwa	Castellano
Ñukapak	Mi, mio.
Kikinpak	Su, suyo/a.
Kanpak	Tú, tuyo/a.
Paypak	Su, suyo, de aquel o aquello, de él o ella.
Ñukanchipak	Nuestro/a.
Kankunapak	Vuestro, de ustedes.
Paykunapak	De ellos/as, de aquellos/as.

1.1.4. La palabra.

Las palabras en Kichwa están constituidas por lexemas y morfemas.

1.1.4.1. Clases de palabras en Castellano y Kichwa.

Tabla 1. *Clases de palabras del Castellano y el Kichwa*

Kichwa		Castellano
Verbos	=	Verbos
Sustantivos	=	Sustantivos
Adjetivos	=	Adjetivos
Adverbios	≠	Adverbios
Pronombres	=	Pronombres
Postposiciones	≈	Preposiciones
No existe	≠	Conjunciones
Artículos	≠	Artículos
Interjecciones	=	Interjecciones

Elaborado por: Gómez Pamela & Freire Paúl

Palabras = lexemas + morfemas

1.1.4.2. Lexema o raíz.

El lexema es una parte invariable de la palabra, que contiene el significado de la palabra es también llamada raíz de la palabra.

Las palabras que tienen el mismo lexema forman una familia de palabras.

1.1.4.3. Morfema o afixo.

El morfema es la parte variable de la palabra que se añade al lexema para completar su significado y formar palabras nuevas.

“Son unidades mínimas que van junto a la palabra, nombre, adjetivo o verbo que modifican a la misma. Se dividen en: prefijos, sufijos e infijos”(Paza, 2011, pág. 20).

1.1.4.3.1. Prefijos.

Son morfemas que van antes y unidas del lexema. En Kichwa solo se tiene este prefijo
Ila

Ejemplos

Kichwa		Castellano
Ilamama	=	madrastra
Ilayaya	=	padrastra
Ilaushushi	=	hijastra

1.1.4.3.2. Sufijos.

Son morfemas que van después y unidas a los lexemas; en el idioma Kichwa son muy importantes los sufijos ya es la parte esencial para la creación de diferentes palabras y existen alrededor de 80 sufijos aquí se tratarán los más importantes para una traducción entendible para el usuario.

Ejemplos

Kichwa		Castellano
Quitomanta	=	desde Quito
Wasimanta	=	de casa
Mikunkapak	=	para comer

1.1.4.3.3. *Infijos.*

Son morfemas que se intercalan en medio de un lexema;

Ejemplos

Kichwa

Castellano

Mikuna

=

comer

Miku**KU**na

=

estar comiendo

1.1.5. Conjugaciones.

Para conjugar un verbo en el idioma Kichwa existe una sola forma de conjugarlo. La raíz del verbo permanece invariable en todos los tiempos, modos y personas; a ello se le añade la terminación verbal respectiva dependiendo del tiempo y persona.

Se tratará 3 tipos de conjugación:

- El presente simple
- El pasado perfecto simple
- El futuro

1.1.5.1. Terminaciones verbales para los tres tiempos.

Tabla 2. Terminaciones verbales del idioma Kichwa según el pronombre

Pronombre	Presente simple	Pasado perfecto simple	Futuro
Ñukaka	-ni	-rkani	-sha
Kikinka	-nki	-rkanki	-nki
Kanka	-nki	-rkanki	-nki
Payka	-n	-rka	-nka
Ñukanchika	-nchik	-rkanchik	-shun
Kankunaka	-nkichik	-rkankichik	-nkichik
Kikinkunaka	-nkichik	-rkankichik	-nkichik
Paykunaka	-nkuna	-rkakuna	-nkakuna

Nota: “El pronombre kanka se utiliza cuando hay confianza con la persona, el pronombre kikin se utiliza por respeto a la persona”(Paza, 2011, pág. 23).

Elaborado por: Gómez Pamela & Freire Paúl

Conjugación: verbo kawsana– vivir

1.1.5.2. Presente simple.

Pronombre	Raiz verbal + terminación - presente	
Ñukaka	kawsani	Yo vivo.
Kikinka	kawsanki	Usted vive.
Kanka	kawsanki	Tú vives.
Payka	kawsan	El/ella vive.
Ñukanchika	kawsanchik	Nosotros/as vivimos.
Kankunaka	kawsankichik	Ustedes viven.

Kikinkunaka	kawsank ichik	Vosotros vivis.
Paykunaka	kawsank kuna	Ellos/ ellas viven.

1.1.5.3. Pasado perfecto simple.

Terminaciones verbales del tiempo pasado perfecto simple.

Pronombre	Raíz verbal + terminación pasado	
Ñukaka	kawsark kani	Yo vivi.
Kikinka	kawsark kanki	Usted vivió.
Kanka	kawsark kanki	Tú viviste.
Payka	kawsark ka	El/ella vivió.
Ñukanchika	kawsark achik	Nosotros/as vivimos.
Kankunaka	kawsark kankichik	Ustedes vivieron.
Kikinkunaka	kawsark kankichik	Vosotros vivisteis.
Paykunaka	kawsark akuna	Ellos/ ellas vivieron.

1.1.5.4. Futuro.

Terminaciones verbales del tiempo futuro

Pronombre	Raíz verbal + tiempo futuro	
Ñukaka	kawsash a	Yo viviré.
Kikinka	kawsank i	Usted vivirá.
Kanka	kawsank i	Tú vivirás.
Payka	kawsank a	El/ella vivirá.
Ñukanchika	kawsash un	Nosotros/as viviremos.
Kankunaka	kawsank ichik	Ustedes vivirán.
Kikinkunaka	kawsank ichik	Vosotros viviréis.

Paykunaka

kawsank**akuna**

Ellos/ ellas vivirán.

1.1.6. La importancia del verbo kana.

Conjugación del verbo kana = ser o estar

Ñukaka	kani	Yo soy, yo estoy.
Kikinka	kanki	Usted es, usted está.
Kanka	kanki	Tú eres, tu estas.
Payka	kan	Él/ella es, él/ella está.
Ñukanchika	kanchik	Nosotros/as somos/estamos.
Kankunaka	kankichik	Ustedes son, ustedes están.
Kikinkunaka	kankichik	Vosotros son, vosotros están.
Paykunaka	kankuna	Ellos/ ellas son, ellos/ellas están.

Ejemplo

Kichwa

Ñukaka wasipi kani

Ñukapak wasipi kani

Ñukaka llamkak kani

Kanka jampikmi kanki

Castellano

Yo estoy en la casa.

Estoy en mi casa.

Yo soy trabajador.

Tú eres médico

Sufijos más importantes

1.1.7. Chu (Interrogativo).

Morfema de interrogación que sirve para hacer preguntas.

Fórmula interrogativa completa:

Pronombre + nombre + -chu + verbo (P + N + -CHU + V)

Ejemplos

Kichwa

Kikinka José**chu** kanki.

Castellano

¿Usted es José?

“El morfema –CHU puede estar en el nombre/pronombre, en el verbo o en el complemento”(Paza, 2011, pág. 30).

Kichwa

Kanpak mamaka shamunk**achu**

Kanpak mam**achu** shamunka.

Kanka wawata charink**ichu**

Castellano

¿Es **de venir** tu mamá?

¿**Tu mamá** es de venir?

¿Tú **tienes** hijo?

1.1.8. Como responder.

Caso 1

Fórmula afirmativa respuesta:

Ari + nombre o pronombre + - mi + verbo (ARI + N/P + -MI + V)

Ejemplos

Kichwa

Ari, José**mi** kani

Ari, ñukaka José**mi** kani

Castellano

Si, soy José

Si, yo soy José

Caso 2

Fórmula negativa respuesta:

Mana + nombre o pronombre + - chu + verbo (MANA + N/P + - CHU + V)

Kichwa

Mana, Joséchu kani.

Mana, ñukaka Joséchu kani

Castellano

No soy José

Yo, no soy José

Caso 3

Fórmula aclarativa respuesta:

Mana + nombre o pronombre + - mi + verbo (MANA + N/P + -MI + V)

Kichwa

Mana. Ñukaka Pedromi kani.

Mana, mana ñukaka Pedromi kani

Mana, ñukaka Joséchu kani. Pedromi kani.

Castellano

No. Yo soy Pedro.

No, Yo no soy Pedro.

Yo, no soy José. Soy Pedro.

1.1.9. Sufijo pish, pash (y, también).

Van junto a los nombres implica inclusión; toma el significado de y, también.

“En Imbabura se pronuncia con el morfema -pash y el resto de las provincias con el - pish”(Paza, 2011, págs. 32,33).

Ejemplo Como inclusión

Kichwa

Ñukapish Quitomantami kani

Ñukaka unkushkami kani, paypash unkushkami
también está enfermo.

Castellano

Yo **también** soy de Quito

Yo estoy enfermo, **y** él

Ejemplo Como conjunción

Kichwa

José**pish** Mañu**pish** yachachikkunami

Castellano

José y Manuel son profesores.

1.1.10. Sufijo manta (de, desde).

Indica procedencia u origen (de, desde).

Kichwa

Ñukaka Colta**manta** kani.

Pedroka Quitom**antami** shamun

Castellano

Yo soy **de** Colta.

Pedro viene **desde** Quito.

1.1.11. Sufijo kama (hasta).

Morfema relacionado al tiempo y el espacio.

Ejemplo tiempo

Kichwa

Kaya**kama**

Ashta**kama**

Castellano

Hasta mañana.

Hasta luego.

Ejemplo espacio

Kichwa

Wasika**mami** rini

Castellano

Voy hasta la casa

1.1.12. Sufijo intin (junto con).

Morfema que indica una estrecha relación ‘junto con’.

Ejemplo

Kichwa

Castellano

Warm**intinmi** shamurkani - vine junto con la mujer.

Tamy**antinmi** purimuni - vengo caminando junto a la lluvia.

1.1.13. Sufijo man (a, hacia).

Morfema que se añade al objeto hacia el que se dirige la acción.

Ejemplos

Kichwa

Castellano

Ñukaka wasi**man** mirini

Yo voy a la casa. (El destino es la casa)

Kinka Quit**oman** churinki

¿Va usted a Quito?

1.1.14. Sufijo wan (con).

Morfema que indica compañía va con un nombre.

Ejemplos

Kichwa

Castellano

Yayaka ishkay allku**wan** purín.

Papá anda **con** dos perros.

Ñañaka turi**wan** shamun.
hermano.

La hermana **viene** acompañada del

1.2. Introducción a la Inteligencia Artificial

“Podemos definir a la inteligencia artificial (IA) como la rama de la ciencia informática dedicada al desarrollo de agentes racionales no vivos, o dicho de otro modo como, al estudio de la manera en la cuales, actualmente, la gente es mejor. De esta manera podemos ver que el entendimiento de algún lenguaje natural, reconocimiento de imágenes, encontrar mejor la manera de resolver un problema de matemáticas encontrar la ruta óptima para llegar a un objetivo específico, etc., son parte del razonamiento humano, y que hasta ahora el hombre ha deseado poder imitarla desarrollando la inteligencia artificial.”(Galeon, 2001).

1.2.1. Principales áreas de la inteligencia artificial.

- Procesamiento de lenguaje natural
- Sistemas expertos
- Robótica
- Programación automática
- Consulta inteligente de base de datos
- Percepción y reconocimiento de patrones
- Autoaprendizaje
- Prueba automática de teoremas y matemáticas simbólica
- Problemas de optimización combinatorios y de itinerarios

1.3. Introducción al procesamiento de lenguaje natural

El procesamiento de lenguaje natural es una rama de la inteligencia artificial que tiene como uno de sus principales objetivos el diseño y la creación de sistemas computacionales que permitan la comunicación entre el hombre y el ordenador a través del lenguaje natural codificándolo en lenguaje computacional.

1.3.1. Historia.

Se crea en el década de los 60, como un subárea de la inteligencia artificial y la lingüística, con el objetivo de estudiar los primeros intentos de traducción de textos mediante un computador; derivados de la generación y comprensión automática del lenguaje natural.

1.3.2. Análisis.

El procesamiento de lenguaje natural se trabaja bajo 4 parámetros, faces o niveles de análisis:

- Morfología
- Sintaxis
- Semántica
- Pragmática

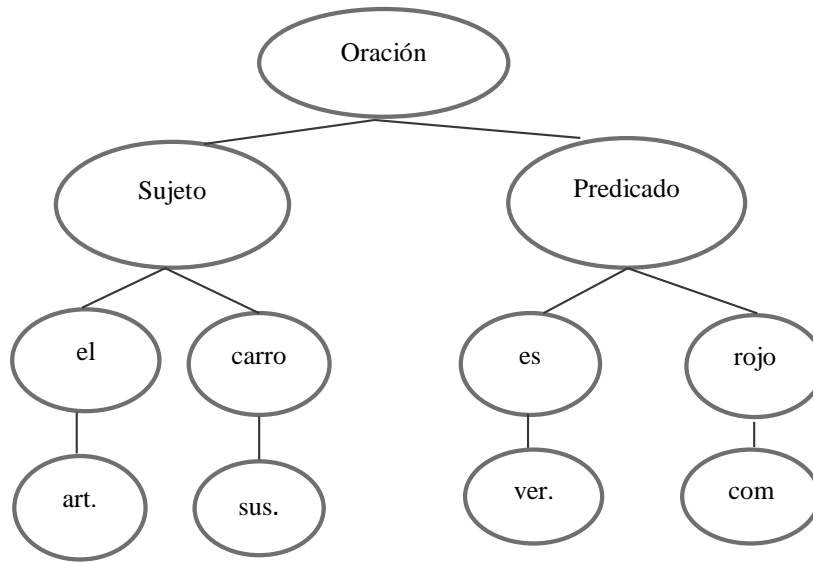
1.3.2.1. Morfología.

Es la parte de la gramática encargada de estudiar la estructura y la forma de las palabras, teniendo como unidad mínima al morfema y a la palabra como unidad máxima.

1.3.2.2. Sintaxis.

Es la parte de la gramática que estudia las distintas relaciones que se dan entre las palabras para poder formar una oración; es decir entender como está formado un texto.

Figura 1. Análisis sintáctico de la oración



Elaborado por: Gómez Pamela & Freire Paúl

1.3.2.3. Semántica.

Es la parte de la gramática que estudia el significado de las palabras.

En el procesamiento de lenguaje natural la semántica es la que se encarga de establecer métodos para que el significado de cada sentencia sea coherente y lógico; estos métodos pueden ser reglas de inferencia.

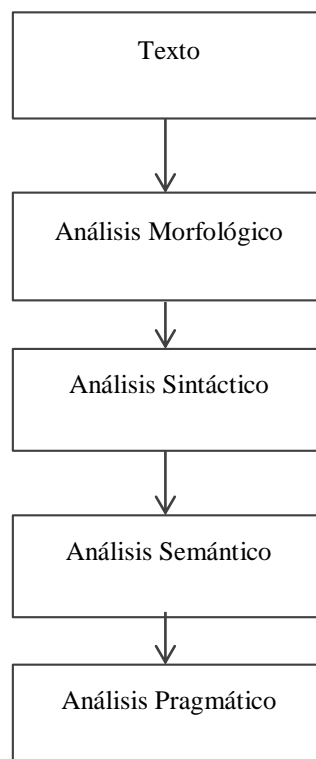
1.3.2.4. Pragmática.

Es la rama de la gramática más compleja ya que complementa al análisis semántico incorporando información para tratar de dar un contexto lógico e interpretar la oración, esto se enfoca en analizar las ambigüedades considerando los datos que se aportan al texto, es decir como el hablante interpreta y produce un mensaje.

1.3.3. Arquitectura.

- El usuario le expresa a la computadora que tipo de procesamiento desea hacer.
- La computadora analiza las oraciones proporcionadas, en el sentido morfológico y sintáctico.
- Luego se analizan las oraciones semánticamente.
- Se realiza el análisis pragmático del texto así se obtiene la expresión final.

Figura 2. Análisis bajo los cuatro parámetros del PLN



Elaborado por: Gómez Pamela & Freire Paúl

1.3.4. Principales dificultades del procesamiento del lenguaje natural.

- Ambigüedades
- Detección de espacios entre palabras
- Recepción imperfecta de datos

1.3.4.1. Ambigüedades.

En lenguaje natural encontramos gran cantidad de ambigüedades, estas ambigüedades se clasifican en:

- **Ambigüedades léxica:** Una misma palabra puede tener distintos significados, por ejemplo juego puedes ser verbo y a la vez sustantivo.
- **Anáforas y catáforas:** Orienta a investigar a que se refiere previamente la oración, para evitar repeticiones innecesarias.
- **Aparición de árboles sintácticos ambiguos:** Una oración puede tener dos interpretaciones distintas.
- **El empleo de ironía, sarcasmo o el uso de distintas entonaciones:** Puede dotar a una misma frase de significados distintos.

1.3.4.2. Detección de espacios entre palabras.

Este tipo de dificultad se da cuando una persona está hablando, ya que se juntan las palabras al vocalizar una frase, pero cuando se trata de procesar texto digitalizado este problema no surge ya que el texto está planteado y no puede existir margen de error entre cada palabra.

1.3.4.3. Recepción imperfecta de datos.

El tema anterior se relaciona en parte con este tema, ya que en ocasiones se recibe frases mal construidas, mal pronunciadas, mal escritas, con fallas ortográficas; para estos casos una persona logrará entender el significado del texto a pesar que se encuentre muy lejano a la expresión correcta, mientras que para una máquina tratara de reconocer el mejor significado dependiendo su conocimiento.

1.3.5. Aplicaciones del procesamiento de lenguaje natural.

- Traducción automática
- Recuperación de información
- Extracción de información y resúmenes
- Resolución cooperativa de problemas
- Tutores inteligentes
- Reconocimiento de voz.

1.3.5.1. Traducción automática.

La traducción automática nace con el deseo de construir máquinas que puedan traducir texto de un lenguaje natural a otro mediante uso de software.

El traducir es una de las artes más complejas ya que no solo se trata de sustituir una palabra con otra si no de reconocer todas las palabras de una frase y el trato que tienen una sobre otra.

1.3.5.1.1. Historia.

- 1951 Instituto de Massachessetts puso a trabajar a YehosshuaBarHellel en la traducción automática.

- 1952 Primer congreso de traducción automática donde se trata los lenguajes controlados, la sintaxis.
- 1954 Primer traductor automático en la universidad de Georgetown (existe ya una traducción automática ya de más de sesenta oraciones del ruso al inglés).
- 1960 Bar Hillel adujo habla ya de incorporar a la semántica y pragmática a la traducción automática para poder tener mejores resultados en la traducción.
- 1976 La universidad de Montreal presenta a METEO sistema que traduce partes meteorológicos del inglés al francés.
- 1980 Se toma mayor interés en modelos estadísticos para la traducción automática.
- En 1991 se da a conocer dos programas de gestión de memorias de traducción, primeramente publicado por IBM (Translation Manager) y posteriormente llevados al gran público por las empresas alemanas TRADOS (Translator's Workbench) y STAR (Transit), y la española ATRIL (Deja-Vu).
- Otro aspecto destacable de esta etapa es el desarrollo de Internet.

En la actualidad se está en una etapa en que la traducción automática es muy importante porque la globalización de empresas y mercados tienen la necesidad de adaptar productos y servicios donde el Internet es pieza clave para una traducción coherente (Artigo, 2012).

1.3.5.1.2. Análisis.

El manejo de la traducción automática se trabaja bajos dos métodos:

- Traducción basada en reglas
- Traducción basada en estadística

➤ **Traducción basada en reglas**

La traducción basada en reglas tiene como base una gran cantidad de reglas lingüísticas integradas de diccionarios bilingües de los idiomas a ser traducidos automáticamente.

El software necesita incluir la morfología, sintáctica y semántica de textos, además varias reglas para poder inferir la estructura gramatical de los idiomas.

“Las traducciones se construyen con diccionarios enormes y reglas lingüísticas sofisticadas. Los usuarios pueden mejorar la calidad de la traducción instantánea añadiendo su terminología al proceso de traducción. Para ello crean diccionarios definidos por el usuario que invalidan la configuración predeterminada del sistema.”(Systran, 2014)

➤ **Traducción basada en estadística**

La traducción automática basada en estadística además, de utilizar las reglas incorporadas en un sistema basados en reglas utiliza modelos de traducción estadísticos cuyos parámetros tratan el análisis de corpus monolingües y bilingües, utilizando diferentes tipos de probabilidad.

Tabla 3. *Traducción automática basada en reglas frente a la estadística*

Basado en reglas	Basado en estadística
Calidad coherente y previsible	Calidad de la traducción imprevisible
Calidad de la traducción fuera del dominio	Calidad fuera del dominio deficiente
Conoce las reglas gramaticales	No conoce la gramática
Alto rendimiento y robustez	Elevados requisitos de CPU y espacio en disco
Coherencia entre versiones	Incoherencia entre versiones
Falta de fluidez	Buena Fluidez
Es difícil manejar la excepciones a la reglas	Buena para tratar la excepciones a las reglas
Costos elevados de desarrollo y personalización	Desarrollo rápido y rentable siempre que exista el corpus requerido

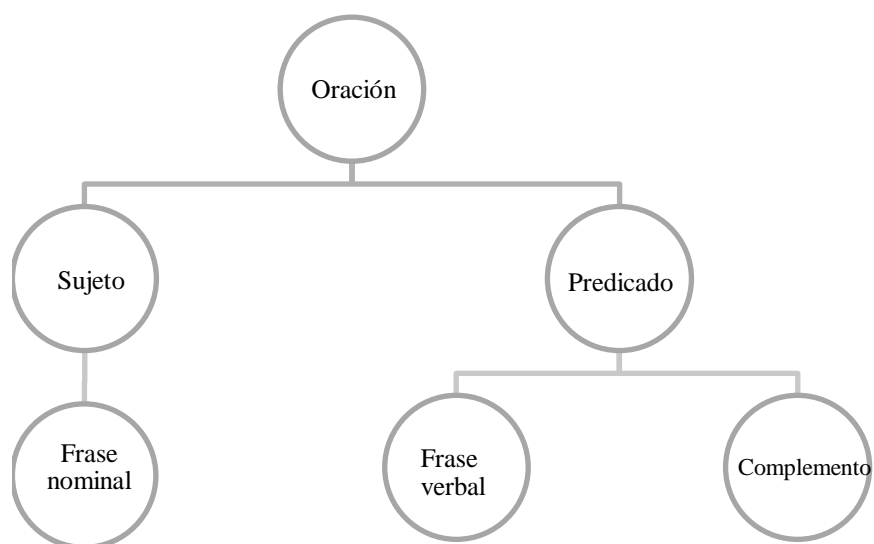
Fuente: Systran 2014

1.3.6. Modelamiento del procesamiento del lenguaje natural

1.3.6.1. Modelamiento de los idiomas Castellano y Kichwa

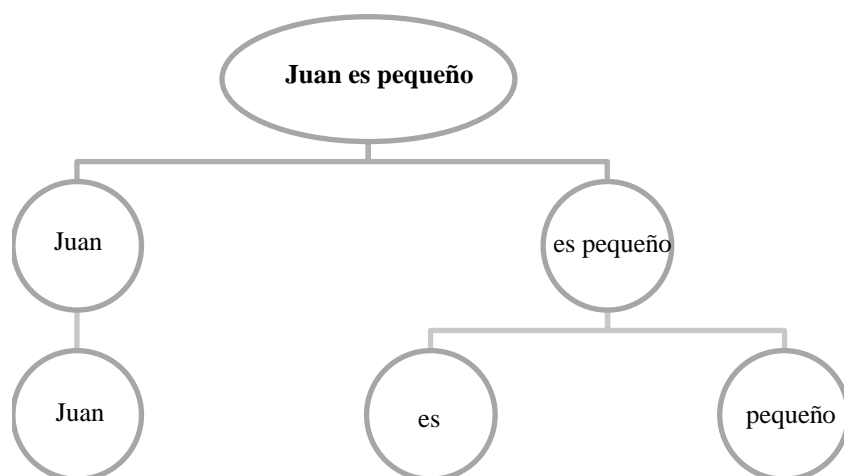
La oración castellano

Figura 3. Modelamiento lenguaje natural la oración Castellano



Elaborado por: Gómez Pamela & Freire Paúl

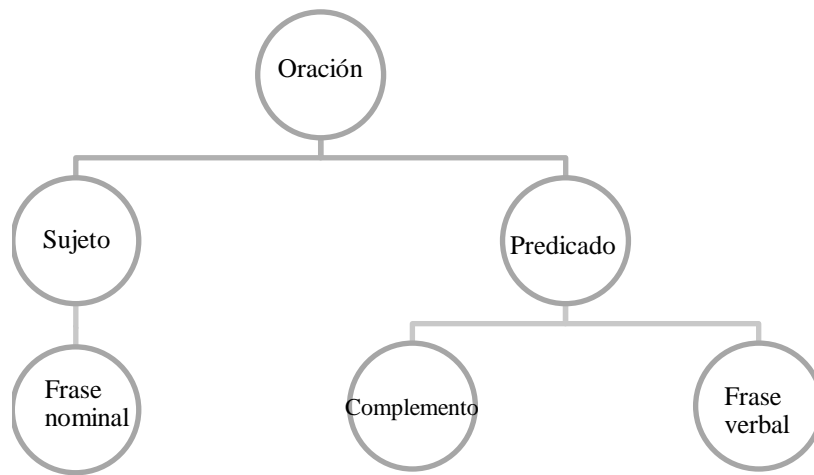
Figura 4. Ejemplo modelamiento lenguaje natural la oración Castellano



Elaborado por: Gómez Pamela & Freire Paúl

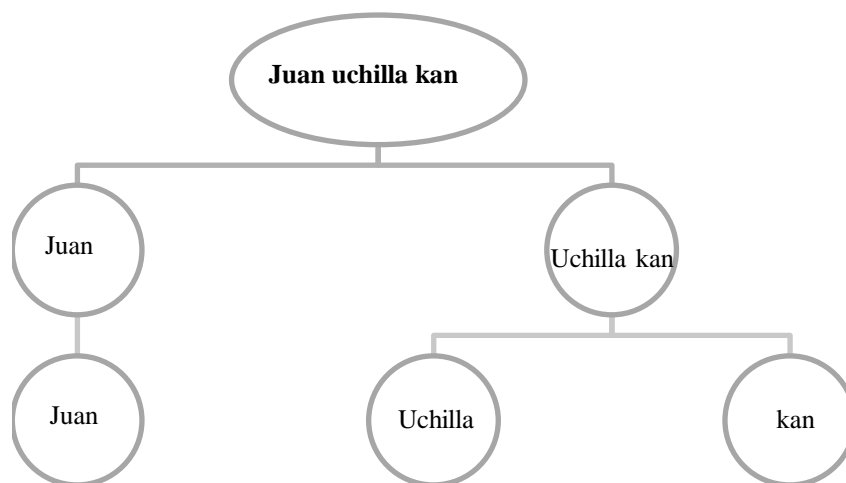
La oración Kichwa

Figura 5. Modelamiento lenguaje natural la oración Kichwa



Elaborado por: Gómez Pamela & Freire Paúl

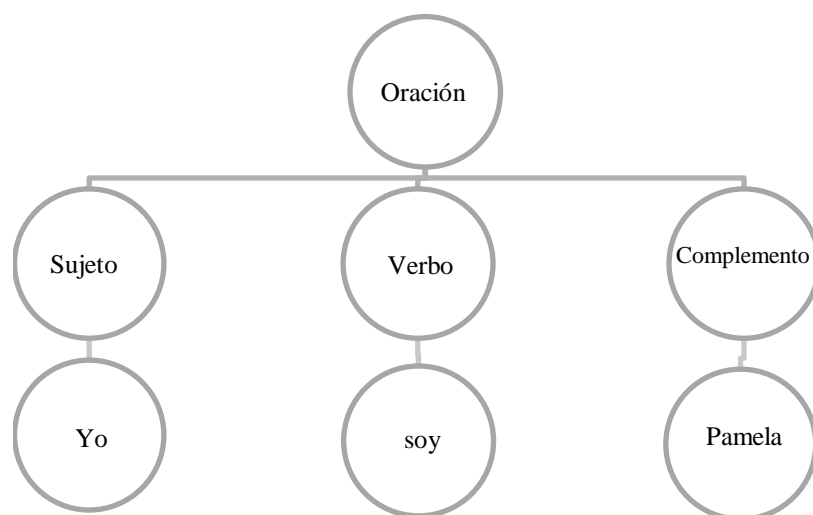
Figura 6. Ejemplo Modelamiento lenguaje natural la oración Kichwa



Elaborado por: Gómez Pamela & Freire Paúl

Pronombre Castellano

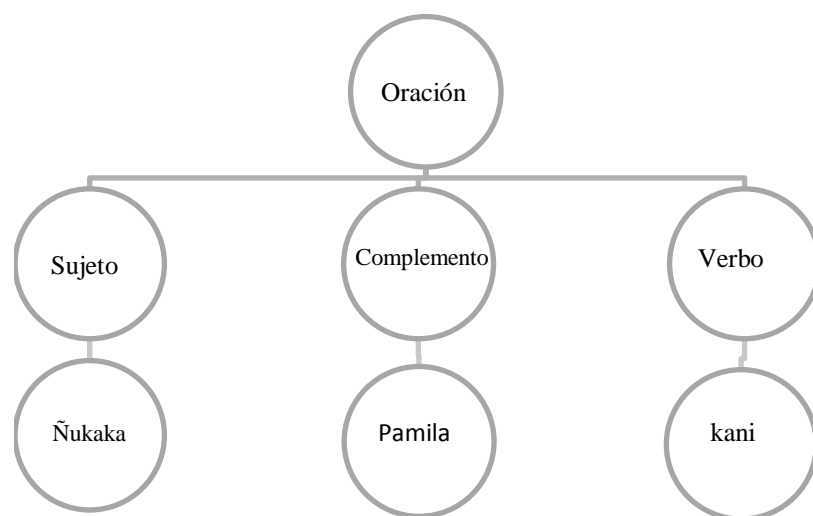
Figura 7. Modelamiento lenguaje natural pronombre Castellano



Elaborado por: Gómez Pamela & Freire Paúl

Pronombre Kichwa

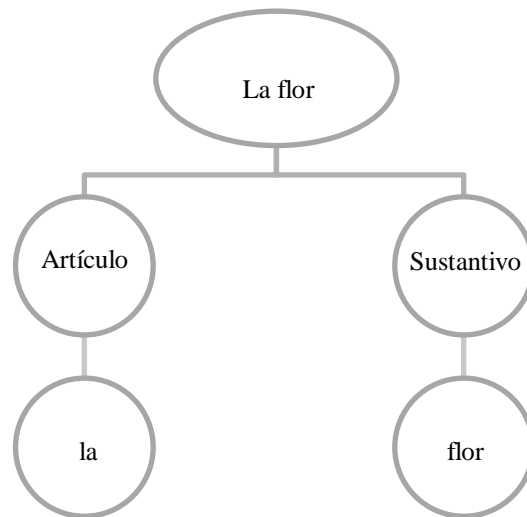
Figura 8. Modelamiento lenguaje natural pronombre Kichwa



Elaborado por: Gómez Pamela & Freire Paúl

Artículo Castellano

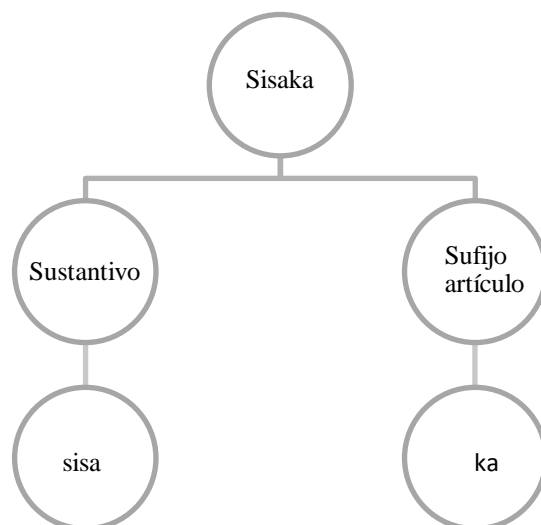
Figura 9. Modelamiento lenguaje natural artículo Castellano



Elaborado por: Gómez Pamela & Freire Paúl

Artículo Kichwa

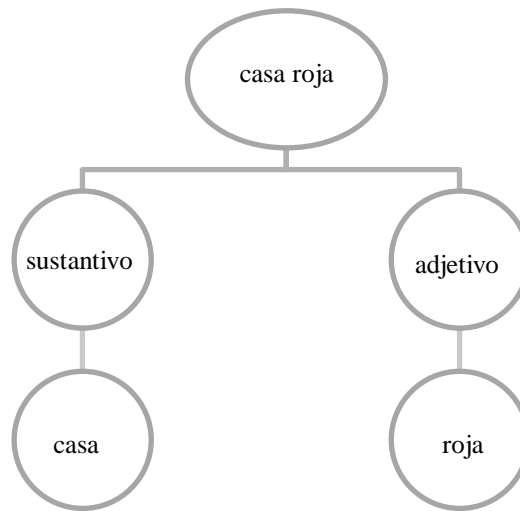
Figura 10. Modelamiento lenguaje natural artículo Kichwa



Elaborado por: Gómez Pamela & Freire Paúl

Adjetivo Castellano

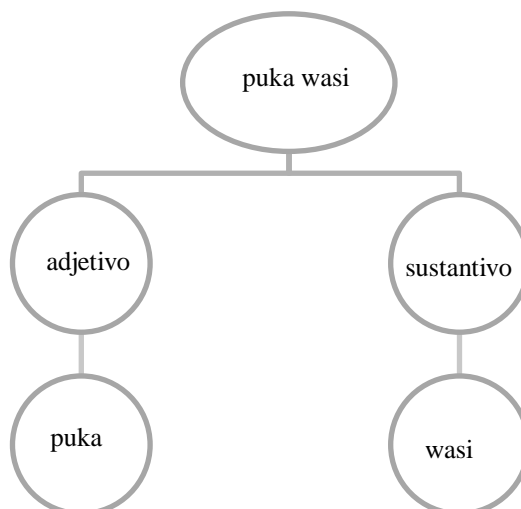
Figura 11. Modelamiento lenguaje natural adjetivo Castellano



Elaborado por: Gómez Pamela & Freire Paúl

Adjetivo Kichwa

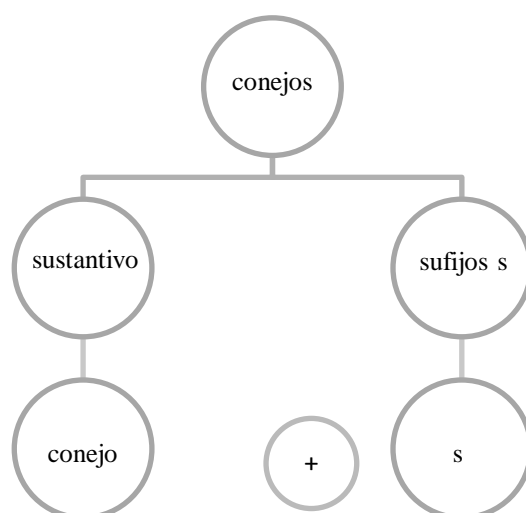
Figura 12. Modelamiento lenguaje natural adjetivo Kichwa



Elaborado por: Gómez Pamela & Freire Paúl

Plural castellano

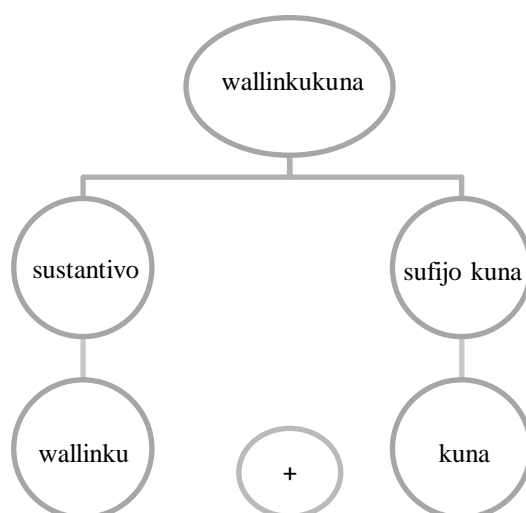
Figura 13. Modelamiento lenguaje natural plural Castellano



Elaborado por: Gómez Pamela & Freire Paúl

Plural Kichwa

Figura 14. Modelamiento lenguaje natural plural Kichwa



Elaborado por: Gómez Pamela & Freire Paúl

1.4. Introducción a los sistemas expertos

1.4.1. Historia.

Los primeros sistemas expertos fueron desarrollados en los años 60 siendo su máximo logro la resolución de sistemas basados en situaciones deterministas, después en los años 70 se empiezan a resolver sistemas basados en situaciones inciertas.

En 1965 Edward Feigenbaum crea el primer sistema experto llamado DENDRAL (estructurado para estudiar un compuesto químico).

En 1972 la universidad de Standford se desarrolló MYCIN para consulta y diagnóstico de infecciones de la sangre.

En esa época se desarrollaron también: HERSAY, que intentaba identificar la palabra hablada, y PROSPECTOR, utilizado para hallar yacimientos minerales.

En 1972 aparece el lenguaje PROLOG.

En 1976 aparece AM (matemático artificial).

En 1979 aparece XCON (servía para configurar todos los ordenadores que salían de la DEC).

En 1980 aparece DELTA (reparación de locomotoras diésel y eléctricas).

A partir de los 90 se desarrolla la inteligencia artificial y por ellos los sistemas expertos toman fuerza como una herramienta habitual en determinadas empresas (Giarratano & Gary, 2001, pág. 30).

1.4.2. Definiciones de sistemas expertos.

Sistema experto es un sistema de computadora que tiene la capacidad de tomar decisiones como un experto humano en un área específica.

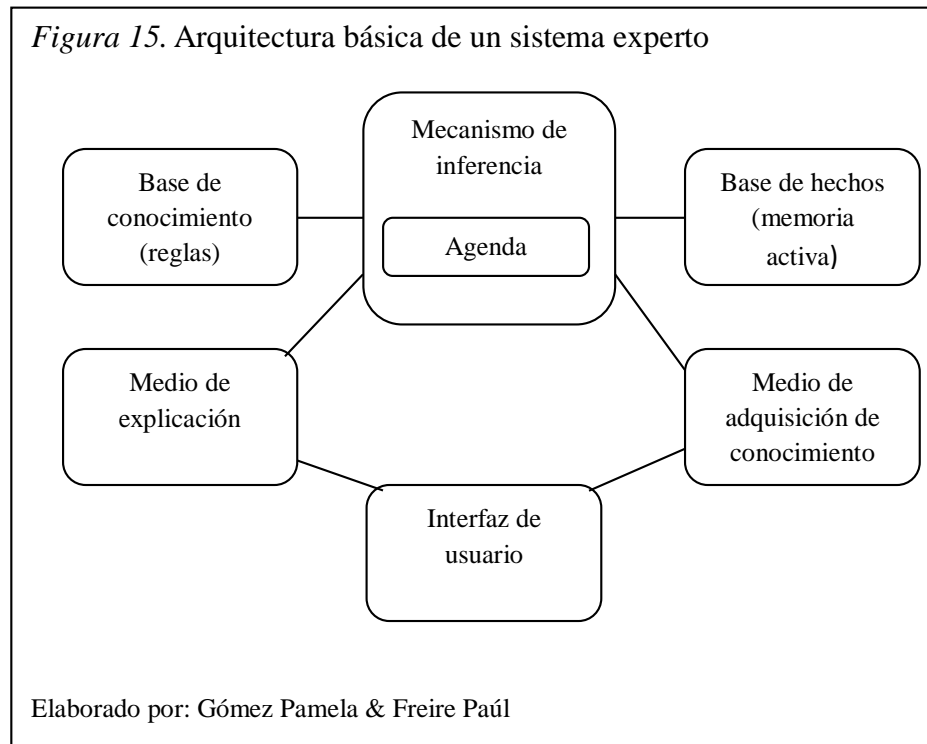
“Un programa de ordenador inteligente que utiliza el conocimiento y los procesos de inferencia para resolver problemas que son lo suficientemente complicados como para requerir de forma significativa la aptitudes de un experto humano para la solución.”(Edward, 1965)

1.4.3. Estructura básica de un sistema experto.

El sistema experto está compuesto por:

- Base de conocimiento
 - Base de hechos
 - Motor de inferencia
 - Módulo de justificación
 - Interfaz de usuario
-
- Base de conocimiento: Es la parte del sistema experto que contiene el conocimiento modelado extraído del experto, está representado mediante reglas.
 - Base de hechos: También llamada memoria de trabajo ya que contiene los hechos sobre un problema que se ha descubierto durante una consulta.
 - Motor de inferencia: es el módulo donde se realiza proceso de razonamiento humano. Este motor de inferencia trabaja con la información contenida en la base de conocimientos y la base de hechos para poder obtener las conclusiones al problema estructurado.
 - Módulo de justificación: Es el módulo donde se explica el razonamiento utilizado por el sistema.
 - Interfaz de usuario: Es la interacción entre el sistema experto y el usuario se realiza en lenguaje natural.

1.4.4. Arquitectura de un sistema experto.



1.4.5. Ventajas y limitaciones de los sistemas expertos.

1.4.5.1. Ventajas.

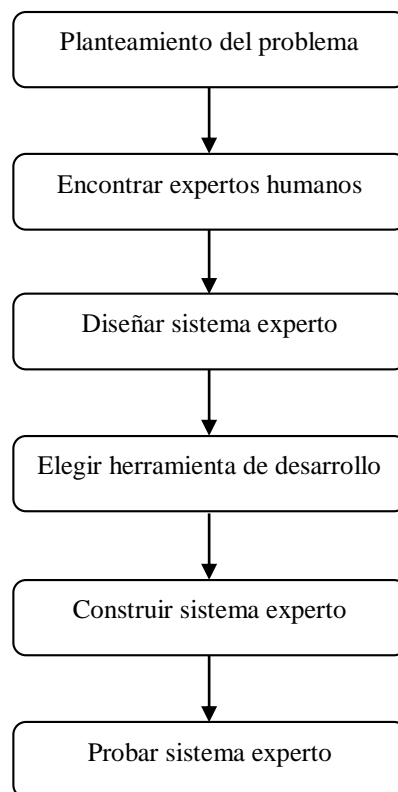
- Permanencia
- Duplicación
- Rapidez
- Bajo costo
- Entornos peligrosos
- Apoyo académico

1.4.5.2. Limitaciones.

Un sistema experto no tiene:

- Sentido común
- Lenguaje natural
- Capacidad de aprendizaje
- Flexibilidad
- Conocimiento no estructurado

Figura 16. Etapas para el desarrollo de un sistema experto



1.4.6. Tipos de sistemas expertos.

Existen principalmente dos tipos de sistemas expertos:

- **Deterministas:** también llamados sistemas expertos basados en reglas.
- **Estocásticos:** también llamados sistemas expertos basados en probabilidad y la estrategia de razonamiento se conoce como “razonamiento probabilístico” o “razonamiento aproximado”.

1.4.6.1. *Sistemas expertos basados en reglas.*

1.4.6.1.1. *Antecedentes.*

Los sistemas expertos basados en reglas son una solución eficiente para resolver muchas situaciones complejas basadas en reglas deterministas, en base a la aplicación reglas y comparación de resultados, siendo las reglas deterministas una de las metodologías más sencillas usadas en los sistemas expertos.

- Base del conocimiento: “En los sistemas basados en reglas intervienen dos elementos importantes: la base del conocimiento y los datos. Los datos están formados por la evidencia o los hechos conocidos en una situación particular. Este elemento es dinámico, es decir, puede cambiar de una aplicación a otra. Por esta razón no es de naturaleza permanente y se almacena en la memoria de trabajo” (Castillo, Gutiérrez, & Hadi, 2000).

Hechos

- Son conocidos en una situación particular.
- Son dinámicos, pueden cambiar de una aplicación a otra.
- Su naturaleza no es permanente.
- Se almacenan en la memoria de trabajo.

Reglas

- En situaciones deterministas, relaciones generales entre conjunto de objetos, y reglas que gobiernan las relaciones.
- Son estáticas, no cambian de una aplicación a otra, a menos que se incorporen al sistema experto elementos de aprendizaje.
- La información es de naturaleza permanente.
- Almacenada en la base de conocimiento

1.4.6.1.2. Definición de reglas.

Una regla es una afirmación lógica que relaciona dos o más objetos e incluyen dos partes: la premisa y la conclusión. Si premisa, entonces conclusión.

- Motor de inferencia: En este módulo trabajan conjuntamente los dos tipos de elementos: los datos (hechos o evidencia) y el conocimiento (conjunto de reglas almacenado en la base del conocimiento).

Para que trabaje el motor de inferencia existen dos mecanismos básicos de inferencia:

- Encadenamiento hacia adelante
- Encadenamiento hacia atrás

1.4.6.1.3. Encadenamiento hacia adelante.

Un motor de inferencia con encadenamiento hacia adelante:

- Parte de los hechos (datos de la memoria de trabajo).
- Va emparejando los datos de la memoria de trabajo con los antecedentes de las reglas.
- Luego las va disparando hasta que se satisface algún objetivo o hasta que ninguna regla se aplicable.

1.4.6.1.4. Encadenamiento hacia atrás.

Un motor de inferencia con encadenamiento hacia atrás:

- Parte de los hechos (datos) y de un objetivo inicial.
- Va emparejando los consecuentes de las reglas con la lista de objetivos.
- Y va aplicando las reglas hacia atrás

1.4.6.2. Sistemas expertos basados en probabilidad.

1.4.6.2.1. Antecedentes.

Los sistemas expertos basados en probabilidad son muy importantes ya que trabaja con la incertidumbre, mientras que los sistemas expertos en reglas son tratados en forma deterministas, en la mayor parte de aplicaciones, la incertidumbre es lo común puesto que los hechos o datos pueden no ser conocidos con exactitud y el conocimiento no es determinista.

Factores para dar lugar a la incertidumbre:

- Información incompleta
- Información errónea
- Información imprecisa

En los primeros sistemas expertos que se hicieron eco de este problema se eligió la probabilidad como medida para tratar la incertidumbre.

1.4.6.2.2. Sistemas expertos basados en razonamiento aproximado.

Los sistemas expertos basados en razonamiento aproximado trabajan con un conocimiento incierto o impreciso y como resultado dará un resultado aproximado y no exacto; es decir no un resultado óptimo sino satisfactorio.

El conocimiento que se necesita para desarrollar los sistemas expertos basados en probabilidad maneja la incertidumbre esta se da mayormente por la deficiencia de información, datos imprecisos, información incompleta, contradictoria y o todo lo que se ingresa no es confiable es por ello que debe tener la capacidad de razonar aproximadamente.

“En el siglo XVIII, Bayes y Laplace propusieron la probabilidad como una medida de la creencia personal hace 200 años. A principios del siglo XX surgen las interpretaciones de la probabilidad como la frecuencia(a largo plazo) asociada a situaciones o experimentos repetibles; en esta línea, destacan especialmente los trabajos estadísticos de Fisher. A principios de los años 30, en cambio, debido sobre todo a los trabajos de L. J. Savage y B. de Finetti, entre otros muchos, se redescubre la probabilidad como medida de la creencia personal. Unos años más tarde, se inventan las computadoras y poco después surge la inteligencia artificial (suele tomarse como punto de referencia el año 1956, en que se celebró la Conferencia de Darmouth”.(Díez, 2005, pág. 3)

En paralelo con esta evolución histórica de crisis y resurgimiento de la probabilidad, se desarrolló la teoría de los conjuntos difusos, frecuentemente llamada con cierta impropiedad lógica difusa.

Para tratar la incertidumbre manejaremos cuatro temas básicos del razonamiento aproximado.

- Método de probabilidad clásica
- Redes bayesianas

- Modelo de factores de certeza MYCIN
- Lógica difusa

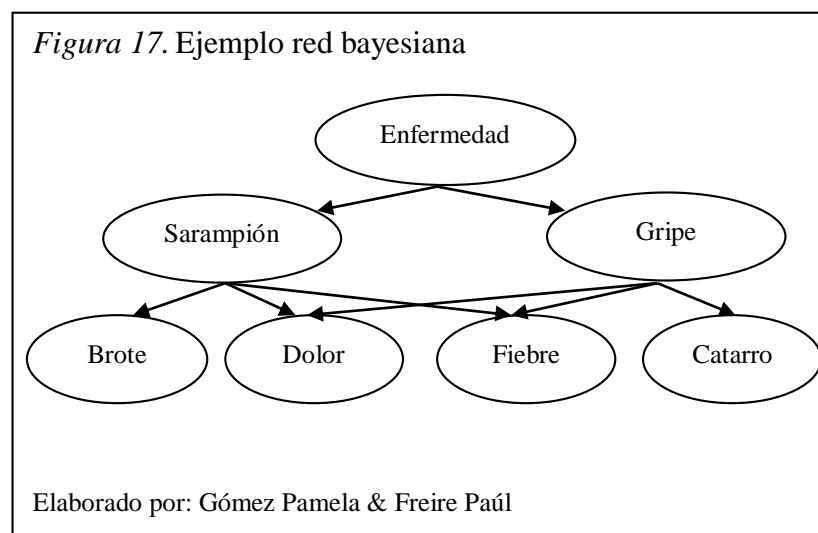
➤ Método de probabilidad clásico

Este método se basa en el teorema de Bayes que propone a la probabilidad como una medida de creencia personal, pero existen algunos inconvenientes para manejar esta información como es la gran cantidad de probabilidades existentes, gran complejidad para razonar. Bajo este sistema se construyeron los primeros sistemas expertos de diagnóstico médico.

➤ Redes bayesianas

“Las redes bayesianas modelan un fenómeno mediante un conjunto de variables y las relaciones de dependencia entre ellas. Dado este modelo, se puede hacer inferencia bayesiana; es decir, estimar la probabilidad posterior de las variables conocidas, en base a variables conocidas.”(Sucar, 2010)

Las redes bayesianas son conjuntos de grafos dirigidos que representan las dependencias para razonamiento probabilístico, en el cual los nodos son variables aleatorias y las aristas son las relaciones de dependencia directa entre las variables.



➤ **Modelo de factores de certeza MYCIN**

Fue creado por Shorliffe & Buchanan en 1975; es un sistema experto (para el diagnóstico de enfermedades infecciosas) tomado como base para los factores de certeza, pero tiene muchas falencias y no son muy utilizadas, ya que carece de base teórica sólida, es muy intuitivo y poco costoso computacionalmente; parte de un sistema basado en reglas con encadenamiento hacia adelante; asocia a cada regla y hecho un número (factor de certidumbre).

Existen dificultades para la toma de datos ya que existe información de naturaleza claramente no estadística, los expertos se resisten a expresar sus procesos de razonamiento en forma probabilística; para la probabilidad se requiere gran cantidad de datos, sujetos a restricciones de independencia; una misma evidencia apoya simultáneamente a una hipótesis y su negación.

➤ **Lógica difusa**

La lógica difusa da sus inicios en 1965 por Lofti A. Zadeh, profesor de la Universidad de California en Berkeley; y fue diseñada para imitar el comportamiento humano, esta metodología proporciona una manera de obtener una conclusión a partir de información vaga, ambigua e imprecisa; es decir la lógica difusa imita como una persona toma decisiones basada en información con las características antes mencionadas.

“La lógica difusa en comparación con la lógica convencional permite trabajar con información que no es exacta para poder definir evaluaciones convencionales, contrario con la lógica tradicional que permite trabajar con información definida y precisa”(Catarina, 2010)

CAPÍTULO 2

ANÁLISIS, DISEÑO Y CONSTRUCCIÓN DEL SISTEMA

2.1. Introducción

Este capítulo definirá entre otras, una de las partes más importantes de un sistema experto como es la adquisición y el modelamiento del conocimiento, lo cual depende en gran parte del experto humano y su interacción con el ingeniero del conocimiento, resultado de esta interacción es el buen funcionamiento de mencionado sistema.

El sistema será capaz de realizar una traducción automática aproximada de texto escrito en Castellano a Kichwa ecuatoriano, el usuario va a interactuar con una interfaz sencilla, la cual solamente requerirá el ingreso del texto en Castellano e inmediatamente después de pulsar el botón traducir, desplegará el texto traducido al idioma Kichwa ecuatoriano.

2.2. Análisis

El análisis del sistema experto es muy importante, ya que determina los componentes necesarios para cumplir los objetivos propuestos del sistema.

2.2.1. Análisis de viabilidad.

Para desarrollar este sistema, en el capítulo del análisis se han considerado dos puntos importantes como son: la viabilidad técnica y económica.

Cabe mencionar que el sistema no será implementado, razón por la cual no se ha considerado una viabilidad operacional.

2.2.1.1. Viabilidad técnica.

Este sistema será desarrollado en Microsoft Visual Studio Express 2012 sobre la plataforma Microsoft Windows, ya que permite el desarrollo de aplicaciones a modo de software libre para estudiantes.

Dado que el sistema no será implementado, no es necesario la obtención de licencias de comercialización del producto.

Para el desarrollo del sistema que comprende entre otros a las pruebas y base de datos, se tiene disponible como herramienta de hardware un computador portátil cuyas características son:

Tabla 4. *Detalle especificaciones de hardware*

Descripción	Cantidad
Procesador: Intel CORE i5	1
Memoria RAM: 4 GB	1
Disco: 450 GB	1
Sistema Operativo: Microsoft Windows 7 de 32 bits	1

Elaborado por: Gómez Pamela & Freire Paúl

2.2.1.2. Viabilidad económica.

La financiación del proyecto requiere principalmente el pago de un curso básico del idioma Kichwa por parte de los tesisas, y entre otras se incluye los gastos operativos que contemplan: transporte, materiales, extras.

A continuación detallamos los valores mencionados:

Tabla 5. *Detalle viabilidad económica*

Descripción	Cantidad	Valor	Subtotal
Curso Kichwa	2	150	300
Gastos operativos	1	300	300
Subtotal			600

Elaborado por: Gómez Pamela & Freire Paúl

Después de analizar los puntos detallados anteriormente se concluye que el sistema es viable para su desarrollo.

2.2.2. Análisis de requerimientos.

A continuación se describirá las características, requerimientos de software, interfaz y la funcionalidad del sistema.

2.2.2.1. Prospectiva del sistema.

El sistema en desarrollo dispondrá de una interfaz sencilla a través de una página de un navegador web, esta interfaz estará directamente enlazada a varios módulos, entre ellos una base de hechos, la que maneja el sistema experto representando los conocimientos relacionados con palabras en idioma Castellano, con la traducción al idioma Kichwa, siendo los mismos conocimientos que maneja el experto humano traductor.

2.2.2.2. Funciones del sistema.

Como se mencionó anteriormente el sistema va a ser desarrollado como aplicación web, el usuario accederá al sistema a través de un navegador web el cual desplegará la interfaz de usuario que se compone de varios módulos como son:

- **Módulo 1:** En este módulo el usuario digitará el texto en idioma Castellano, a traducir al idioma Kichwa ecuatoriano.

Además en este módulo se desplegará el texto traducido al idioma Kichwa ecuatoriano, utilizando solamente el método de traducción basado en reglas de inferencia, además se podrán visualizar las palabras que presenten ambigüedad léxica, dando alternativa al usuario a escoger la opción más acertada acorde a su necesidad, también mediante la mencionada selección de opciones, el sistema calculará la probabilidad de aparición de cada palabra de manera que el sistema aprenderá de este cálculo, siendo esto parte del razonamiento aproximado que realiza el sistema en el siguiente módulo.

- **Módulo 2:** En este módulo se desplegará el texto traducido al idioma Kichwa ecuatoriano, utilizando el método de traducción basado en reglas de inferencia conjuntamente con el método de traducción basado en razonamiento aproximado, siendo mencionado texto el resultado de un gran proceso de inferencia combinado con el aprendizaje adquirido por parte del sistema, por lo cual el texto traducido será lo más coherente posible en base a todo el conocimiento del sistema.

Además en esta sección se desplegará el módulo de justificación, llamado también interfaz explicativa, donde el sistema demostrará claramente que reglas se utilizaron al momento de realizar la traducción, conjuntamente con datos importantes que faciliten la explicación del resultado obtenido..

Cabe mencionar que el sistema no realizará un proceso de autenticación ya que no es un producto de administración, y la información que proporcionará estará disponible al público en general.

2.2.2.3. Restricciones.

El sistema operativo sobre el cual trabajará el sistema deberá ser Microsoft Windows y su acceso deberá ser necesariamente a través de un navegador web; También es necesario mencionar que el usuario deberá apegarse en lo posible a las reglas de escritura del idioma Castellano, es decir tomando muy en cuenta signos de puntuación y ortografía, al momento de ingresar texto a traducir al sistema ya que de esto dependerá en parte la coherencia de la traducción.

2.2.3. Requerimientos funcionales.

Para la especificación de los requerimientos funcionales del sistema, se tomó en cuenta varias sugerencias de distintos usuarios, acerca de lo necesario al momento de realizar una traducción automática mediante una página web de un idioma a otro, llegando a la conclusión que un sistema traductor de idiomas debe necesariamente cumplir con los requerimientos que detallamos a continuación:

2.2.3.1. Usuario del sistema.

Requerimiento funcional 1.1. Interfaz perfil de usuario

Tabla 6. *Requerimiento funcional: Interfaz perfil de usuario*

Número	01
Requisito	Interfaz perfil de usuario.
Descripción	Desplegar la pantalla de interacción con el actor.
Precondición	Ejecución del sistema.
Entrada	Ninguna.
Proceso	Exploración de la interfaz del sistema.
Salida	Interfaz del sistema amigable al usuario.
Actor	Usuario.
Observación	Ninguna.

Elaborado por: Gómez Pamela & Freire Paúl

Requerimiento funcional 1.2. Traducción coherente

Tabla 7. *Requerimiento funcional: Traducción coherente*

Número	02
Requisito	Traducción coherente.
Descripción	Permitir que el actor confíe en la coherencia del resultado de la traducción.
Precondición	Ninguna.
Entrada	Texto en idioma Castellano.
Proceso	El actor digitará texto en castellano a traducir, tomando en cuenta las reglas generales del idioma.
Salida	Traducción de manera coherente y entendible.
Actor	Usuario.
Observación	El actor deberá tomar en cuenta signos de ortografía y puntuación al momento del ingreso de texto.

Elaborado por: Gómez Pamela & Freire Paúl

Requerimiento funcional 1.3. Traducción con detección de ambigüedades léxicas

Tabla 8. *Requerimiento funcional: Traducción con detección de ambigüedades léxicas*

Número	04
Requisito	Traducción con detección de ambigüedades léxicas.
Descripción	Detectar ambigüedades léxicas en el módulo de traducción basado en reglas de inferencia.
Precondición	Ninguna.
Entrada	Texto en idioma Castellano.
Proceso	El actor digitará texto en Castellano a traducir, en el módulo de traducción basado solamente en reglas de inferencia.
Salida	Detección de las palabras encontradas que contengan ambigüedad léxica.
Actor	Usuario.
Observación	En este módulo el sistema solamente detectará ambigüedades léxicas, presentando aun así una traducción coherente.

Elaborado por: Gómez Pamela & Freire Paúl

Requerimiento funcional 1.4. Traducción sugerida

Tabla 9. *Requerimiento funcional: Traducción sugerida*

Número	05
Requisito	Traducción sugerida.
Descripción	Solucionar ambigüedades léxicas.
Precondición	Ninguna.
Entrada	Texto en idioma Castellano.
Proceso	El actor pulsará un botón con la etiqueta “Traducción Sugerida”, en el módulo de traducción basado en reglas de inferencia y razonamiento aproximado.
Salida	Solución a gran parte de las ambigüedades léxicas detectadas en el módulo de traducción basado en reglas de inferencia por parte del conocimiento del sistema.
Actor	Usuario.
Observación	Ninguna.

Elaborado por: Gómez Pamela & Freire Paúl

Requerimiento funcional 1.5. Traducción en presente, pasado y futuro

Tabla 10. *Requerimiento funcional: Traducción en presente, pasado y futuro*

Número	06
Requisito	Traducción en presente, pasado y futuro.
Descripción	Traducir texto sin importar el tiempo en que se encuentre.
Precondición	Ninguna.
Entrada	Texto en idioma Castellano en los tiempos presente, pasado y futuro.
Proceso	El actor ingresará texto en presente, pasado y futuro indistintamente al sistema
Salida	Traducción de manera normal presentando coherencia en los distintos tiempos en que se encuentre el texto ingresado.
Actor	Usuario.
Observación	Ninguna.

Elaborado por: Gómez Pamela & Freire Paúl

2.3. Diseño

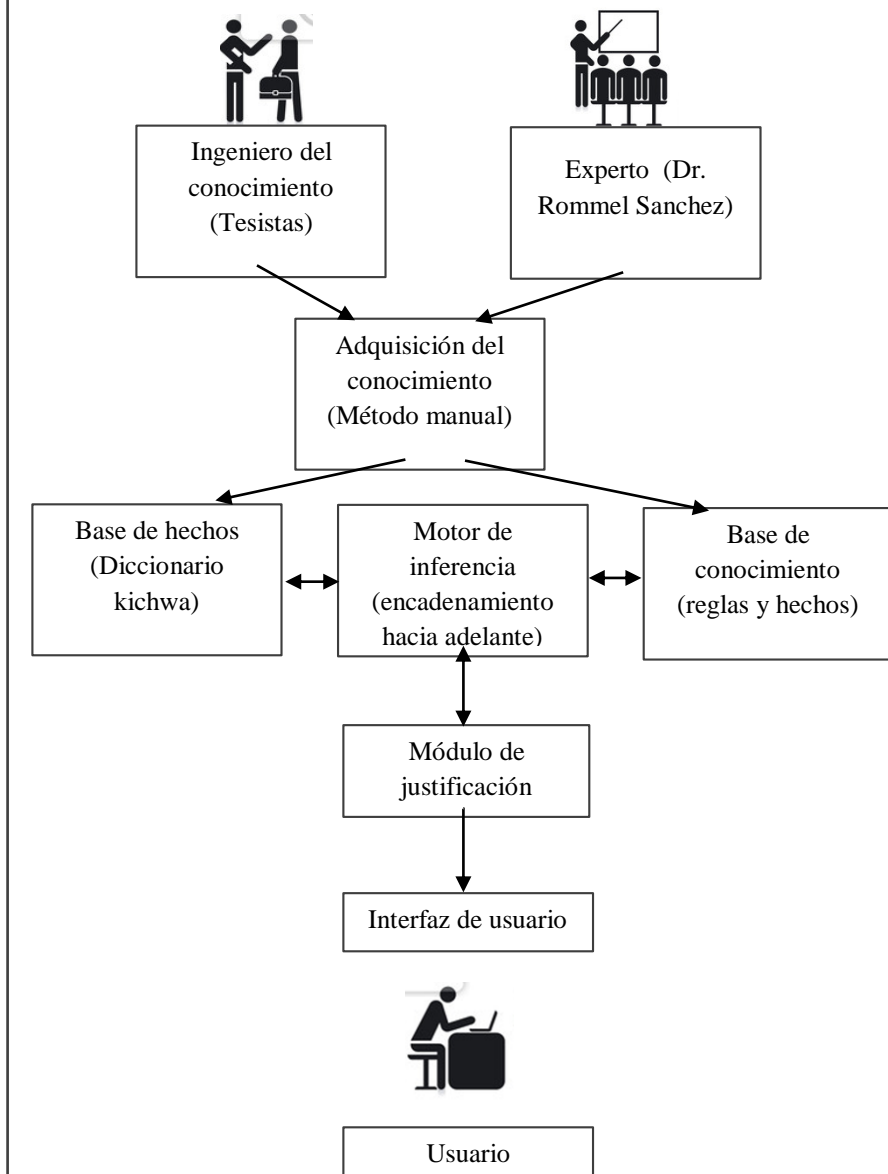
A continuación se definirá el diseño con cada uno de los actores del sistema con sus respectivas funciones. Se representará la estructura que tendrá el sistema, y se diseñará cada uno de los módulos componentes de la aplicación para que el usuario pueda aprovechar de la funcionalidad que ofrecerá.

2.3.1. Arquitectura del sistema experto.

Para el desarrollo del sistema se ha tomado en cuenta todos los elementos necesarios que interactuando entre si forman la aplicación web traductor, siendo entre de los más importantes el elemento usuario, al cual se lo denomina también cliente web, el elemento base de hechos, al cual se lo denomina también base de datos.

La siguiente figura muestra la estructura de nuestro sistema experto con sus componentes y los actores necesarios: base del conocimiento, base de hechos, motor de inferencia, la interfaz de usuario, ingeniero del conocimiento, experto y el usuario.

Figura 18. Arquitectura del sistema



Elaborado por: Gómez Pamela & Freire Paúl

2.3.2. Estructura del sistema.

2.3.2.1. Actores.

- Experto
- Ingeniero del conocimiento
- Usuario

2.3.2.1.1. Experto.

El experto es la persona que domina el tema, en este caso el idioma Kichwa ecuatoriano, del cual se extrae toda la información necesaria para ingresarla en el sistema.

Para nuestro sistema contamos con la colaboración del Dr. Rommel Sánchez, quien además de contar con un título en jurisprudencia es profesor del idioma Kichwa en la Universidad Central del Ecuador, característica indispensable para convertirse en el experto de nuestro proyecto.

2.3.2.1.2. Ingeniero del conocimiento.

El ingeniero del conocimiento es la persona capaz de tomar y entender toda la información que le da el experto en forma natural, y mediante varios métodos técnicos, convertirlos en lenguaje computacional e ingresarlos en el sistema.

Además, de lo antes mencionado el ingeniero del conocimiento debe tener la capacidad de interactuar con el usuario, entendiendo sus necesidades para luego satisfacerlas en el diseño del sistema.

En el sistema traductor el papel de ingenieros del conocimiento lo ejecutarán los tesisistas para lo cual fue estrictamente necesario tomar un curso básico del idioma Kichwa ecuatoriano impartido por el antes mencionado experto.

2.3.2.1.3. Usuario.

El usuario es una persona la cual tiene la necesidad de hacer uso de un traductor Castellano – Kichwa ecuatoriano para fines particulares.

2.3.2.2. Conocimiento del sistema.

2.3.2.2.1. Adquisición del conocimiento.

El proceso de adquisición del conocimiento es la labor que tiene el ingeniero del conocimiento para extraer el conocimiento del experto.

➤ Etapas de la adquisición del conocimiento

- **Identificación:** Etapa donde se identifica el problema y sus características. En este caso el problema es cuando se da la necesidad de traducción.
- **Entendimiento:** Entender si el conocimiento es importante para resolver el problema. Para nuestro sistema es muy importante el conocimiento impartido por parte del experto, ya que de ello depende la respuesta del sistema.
- **Formalización:** En esta etapa se formaliza el conocimiento y se lo organiza para que pueda actuar con la base de hechos, es la etapa más difícil ya que escoge el tipo de software y las reglas que se va a utilizar, en base al conocimiento adquirido por parte del experto.
- **Implementación:** Involucra directamente a la programación del conocimiento que se utilizara para el sistema experto, para lo cual se utiliza reglas de inferencia basadas en modus ponens.
- **Pruebas:** El ingeniero del conocimiento prueba los algoritmos con ejemplos para que no existan fallas, para lo cual se ingresa texto indistintamente poniendo a prueba la capacidad de respuesta en diferentes situaciones del sistema.

Para la obtención del conocimiento se escogió el método manual que fundamentalmente está estructurado por medio de entrevistas, donde el ingeniero del

conocimiento obtiene el conocimiento mediante contacto directo con el experto y otras fuentes como conocimiento documentado, para posteriormente codificarlo en la base del conocimiento; para ello fue indispensable la toma de un curso básico del idioma Kichwa por parte de los tesisas, dicho curso fue comprendido en dos niveles en donde se detalla la estructura y gramática del idioma, impartidos por el experto, a partir de ahí se consiguió interactuar con el sistema modelando las reglas del idioma.

2.3.2.2.2. Base del conocimiento.

Contiene los datos que el ingeniero del conocimiento ingresa al sistema, una vez adquirido el conocimiento impartido por el experto en lenguaje natural y modelado por el ingeniero a lenguaje computacional, de tal manera que pueda interactuar la base del conocimiento con la base de hechos, dicho conocimiento se encuentra estructurado mediante reglas de inferencia.

Cada regla está formada de una parte denominada premisa y de una parte denominada conclusión y tendrá la siguiente estructura:

SI premisa ENTONCES conclusión

- Si \exists palabra \Rightarrow \exists hecho
- Si \exists signos de puntuación (. , ;) \Rightarrow oración
- Si \exists hecho \Rightarrow traducción directa
- Si tipo_termino ver \Rightarrow verbo regular \Rightarrow método traduce ()
- Si palabra es primera letra mayúscula \Rightarrow nombre propio
- Si tipo_palabra suf \Rightarrow traduce sufijo \wedge palabra adelante + sufijo

- Si nombre propio \Rightarrow método (cambia letra)
- Si palabra contiene hecho \wedge terminación esta conjugada \Rightarrow es verbo regular
- Si verbo regular \Rightarrow palabra anterior + ta
- Si tipo_palabra es ver \Rightarrow verbo irregular
- Si palabra termina en s o es \Rightarrow plural
- Si plural \Rightarrow traduce palabra + kuna
- Si tipo_palabra es ver \vee vrb \Rightarrow posición 3
- Si tipo_palabra es sujeto \Rightarrow posición 1
- Si pronombre 0 \wedge terminación verbo base presente \Rightarrow terminación NI
- Si pronombre 0 \wedge terminación verbo base pasado \Rightarrow terminación RKANI
- Si pronombre 0 \wedge terminación verbo base futuro \Rightarrow terminación SHA

También es necesario mencionar que de este tipo de reglas está formada la base del conocimiento del sistema, dichas reglas realizan un proceso de inferencia de encadenamiento hacia adelante. Existe una innumerable cantidad de estas reglas, por esta razón se ha expuesto las más importantes que permite explicar su función y estructura, siendo muy similar a esta la composición del resto de reglas de inferencia contenidas en la base del conocimiento de la aplicación; para la representación del conocimiento se escoge el método de inferencia de modus ponens (afirmando afirma).

2.3.2.2.3. Base de hechos.

La base de hechos es también llamada memoria de trabajo la cual se encuentra estructurada con los hechos obtenidos del diccionario del Ministerio de Educación del Ecuador “Kichwa yachakukkunapa shimiyyuk kamu” del DINEIB (Dirección Nacional de Educación Intercultural Bilingüe).

La base de hechos tiene esta estructura:

Diccionario (id_termino, término, tipo_termino, traducción, probabilidad, conteo)

Diccionario (100, flor, sus, sisa)

Donde sí existe un hecho existe una traducción

- Si \exists hecho $\Rightarrow \exists$ traducción
- flor \Rightarrow sisa

Reglas (id_regla, tipo_termino, num_regla, nombre_regla, regla)

- Si \exists traducción $\Rightarrow \exists$ regla

Figura 19. Diagrama base de hechos

Diccionario		
CP	id_termino	int (11)
	termino	text (20)
	tipo_termino	text (20)
	traduccion	text (20)
	probabilidad	text (20)
	conteo	int (11)

Reglas		
CP	id_regla	int (11)
	tipo_termino	text (20)
	num_regla	int (11)
	nombre_regla	text (20)
	regla	text (50)

Elaborado por: Gómez Pamela & Freire Paúl

2.3.2.2.4. Motor de inferencia.

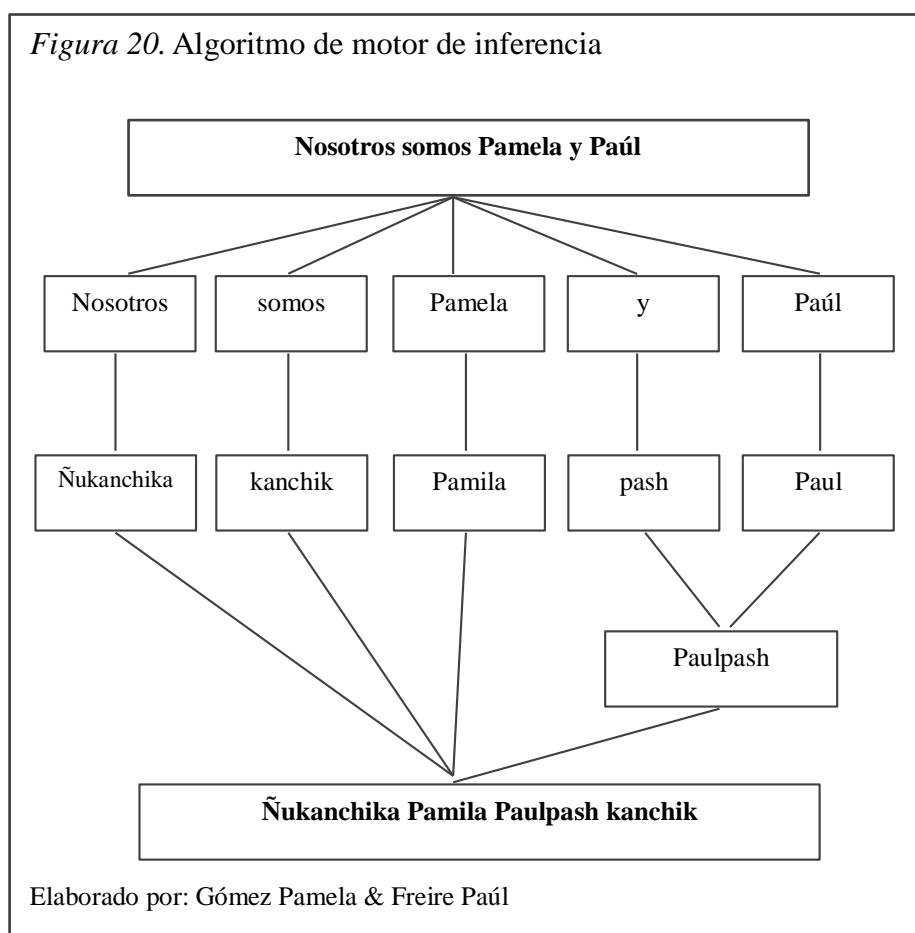
El motor de inferencia es el software que debe tener la capacidad de interactuar con la información que se encuentra en la base de hechos y la base de conocimiento para poder tener resultados acordes a la necesidad del usuario.

La estrategia de inferencia que se utilizará es el “modus ponens”: “Si A entonces B”, se puede concluir que si A es verdadero entonces B es verdadero.

Si \exists palabra $\Rightarrow \exists$ hecho

El método utilizado es un **encadenamiento hacia adelante** también llamado guiado por datos; es decir va recorriendo la base del conocimiento partiendo de los hechos hasta llegar a la conclusión.

Aquí se detallará el algoritmo para la ejecución del motor de inferencia.



➤ **Algoritmo del motor de inferencia**

- Se ingresa al sistema el párrafo.

Nosotros somos Pamela y Paúl.

- Aplica Regla 1

Si \exists signos de puntuación \Rightarrow oración

- Aplica Regla 2

Si \exists oración \Rightarrow separa palabras

- Una vez separadas las palabras; trata a cada palabra para aplicar cada regla.

Ejemplo

Nosotros

Si \exists palabra $\Rightarrow \exists$ hecho

Si \exists hecho \Rightarrow traducción directa

Nosotros \Rightarrow Ñukanchika

- Después de aplicar todas las reglas para cada palabra aplica la regla para ordenar la oración

Si tipo_ palabra es ver o vrb \Rightarrow colocar final de oración.

De esta manera encadena hacia adelante, partiendo de los hechos y recorriendo regla por regla hasta que ya no existan más reglas a ejecutarse y pasar a ordenar oración por oración para presentar el resultado del párrafo final.

2.3.2.3. Interfaz de usuario.

El objetivo principal de una interfaz es ayudar al usuario a proporcionar información necesaria y obtener del sistema lo siguiente:

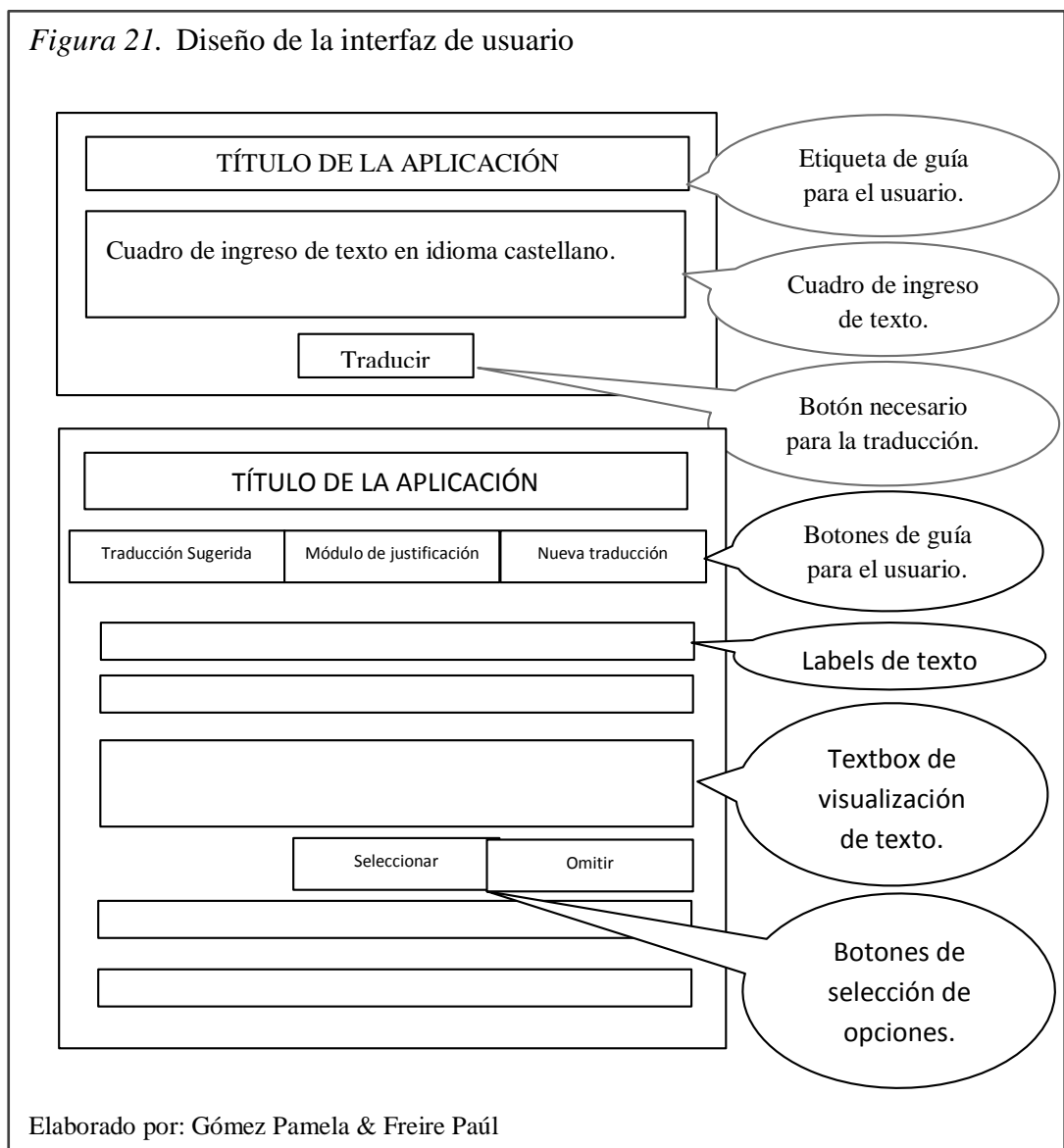
- Debe permitir a los usuarios a acceder al sistema de una forma precisa y acorde con sus necesidades individuales, en este caso de traducción.
- El aumento de la velocidad de captura de datos y reducir errores.
- Consideración al usuario, con el diseño de una interfaz adecuada y que el sistema le proporcione la retroalimentación adecuada.

2.3.2.3.1. Diseño de la interfaz de usuario.

Después de analizar varios traductores existentes en la web y de consultar opiniones de posibles usuarios del sistema, se propuso que para el diseño de la interfaz de usuario se requiere principalmente que la aplicación presente facilidad en su manejo, es decir debe ser amigable con el usuario, y entre otros requisitos debe cumplir con los siguientes:

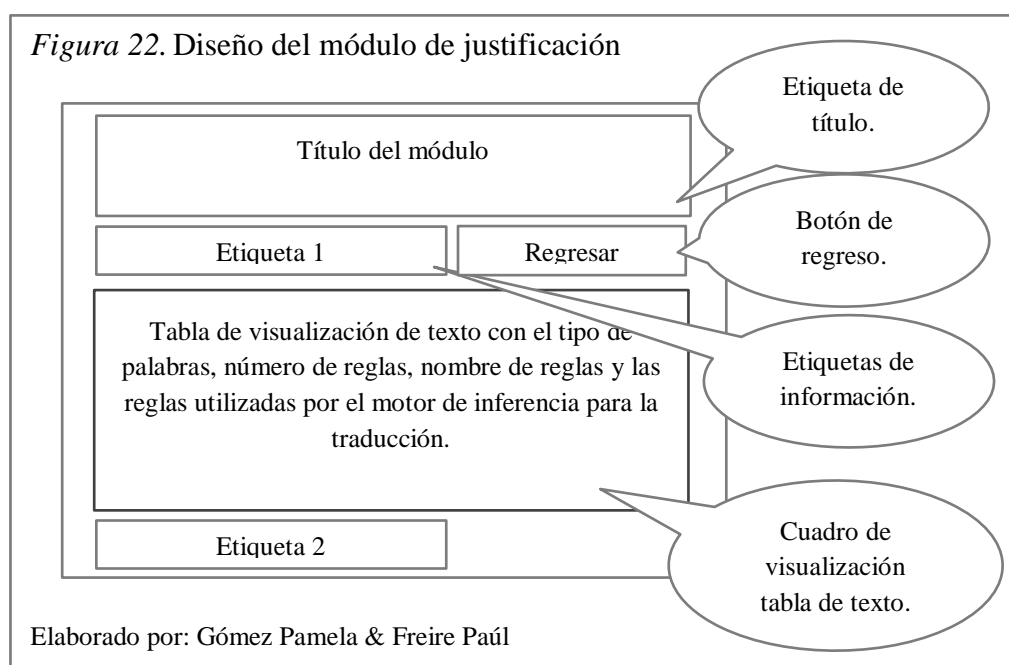
- Presentar un estándar en la ubicación de los botones de comando, con texto claro y relacionado con el tema.
- Los colores utilizados deben ser formales.
- La navegación en la interfaz debe ser a través del mouse y el teclado.
- Se requiere un textbox de entrada, donde el usuario ingresará el texto a traducir y un label de salida, donde después de pulsar un botón con la etiqueta traducir, se presentará el texto traducido.

- Se dará opción al usuario a conocer las palabras a traducir que presenten ambigüedad léxica, y a escoger la opción que más se acerque a su necesidad.
- Se dará opción a mejorar la traducción, para lo cual se requiere un label de salida donde después de pulsar un botón con la etiqueta “Mejorar Traducción”, se desplegará el texto traducido de manera mejorada, con respecto a la traducción visualizada en el label del paso anterior.
- Se necesita desplegar una interfaz explicativa donde se visualice las reglas utilizadas para la obtención de resultados de traducción.



2.3.2.4. Módulo de justificación.

El módulo de justificación es la parte del sistema que explica el razonamiento del mismo, es decir mostrará las reglas utilizadas en el proceso de inferencia del sistema para obtener la traducción, el mismo presentará una pantalla amigable y entendible para el usuario.



2.3.2.5. Cliente web.

Se denomina cliente web al usuario, el mismo que a través de un ordenador accederá a la aplicación por medio de un navegador web.

El sistema operativo instalado para la ejecución del sistema es Microsoft Windows, siendo los navegadores: Internet Explorer o Mozilla Firefox los medios para la ejecución del sistema.

2.3.2.6. Base de datos.

Se ha denominado a MySQL como gestor de base de datos del sistema, y solamente el ingeniero del conocimiento tiene acceso al mismo para modificar información a su conveniencia y en beneficio del crecimiento del sistema. También estará implementado sobre una plataforma Microsoft Windows y tanto la aplicación conjuntamente con la base de datos funcionarán en el mismo ordenador.

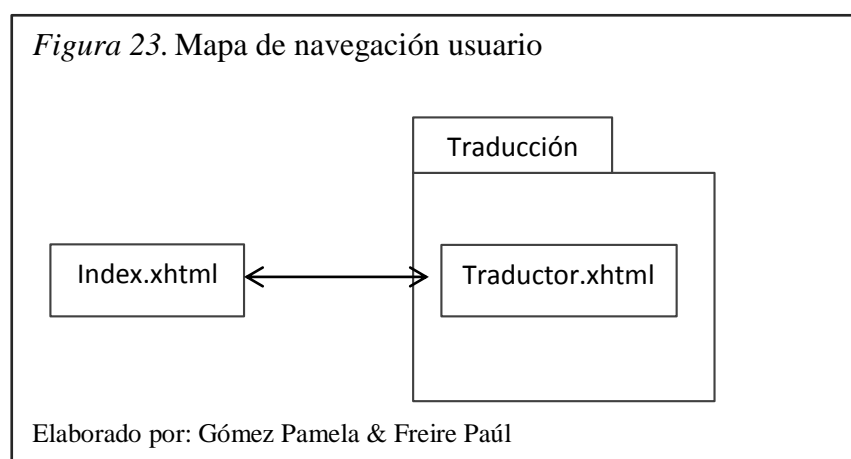
2.3.2.7. Mapa de navegación.

A continuación se expone la manera como la aplicación web permitirá al usuario moverse o desplazarse a través de la misma.

2.3.2.7.1. Mapa de navegación para traducción.

Tras la ejecución del sistema mediante un navegador web, el usuario visualiza la página de traducción, donde se despliegan tres cuadros de texto con sus respectivas etiquetas para guía del usuario y los botones de traducción, cabe mencionar que el diseño muestra una estructura sumamente intuitiva para el usuario, el mismo que podrá hacer uso de la aplicación de forma fácil.

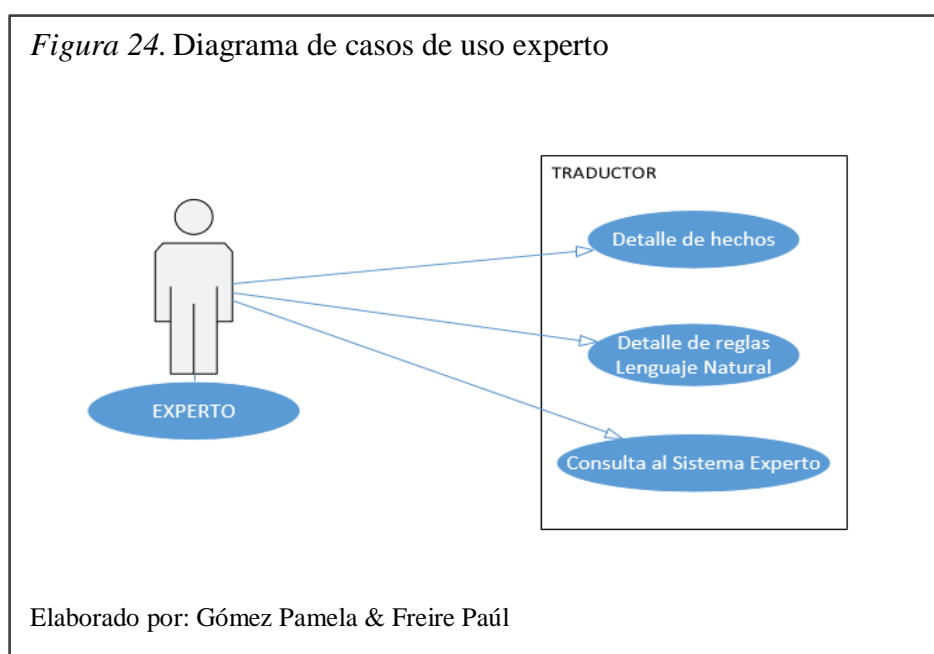
En la siguiente figura se muestra el mapa de navegación para el usuario con el único rol existente que es el de traducción.



2.3.3. Diagramas de casos de uso.

Estos diagramas permitirán especificar y relacionar los actores con los principales módulos componentes del sistema, también facilitarán el diseño mediante la organización de los módulos antes mencionados para un mejor desarrollo.

2.3.3.1. Diagrama de casos de uso experto.



Nombre de caso de uso: Caso de uso experto

Objetivo: Determinar la función del experto

Actor: Experto

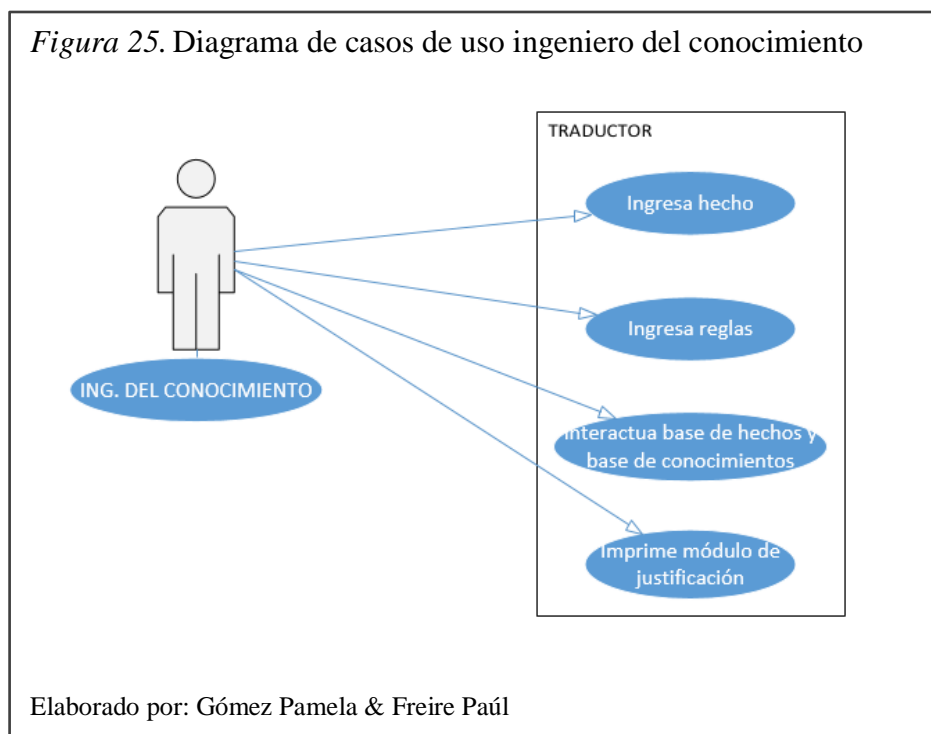
Pre-condición: Ser experto en el idioma Kichwa

Descripción:

El experto tiene la capacidad de consultar la traducción del sistema; la calidad de la traducción depende de los hechos y reglas entregadas por el experto.

Post-condición: El experto puede hacer uso del sistema para visualizar la traducción y así indicar sus sugerencias al ingeniero del conocimiento para que él pueda extender el conocimiento con los detalles que el experto entregue cabe resaltar que el experto entrega el conocimiento en lenguaje natural.

2.3.3.2. Diagrama de casos de uso del ingeniero del conocimiento.



Nombre de caso de uso: Caso de uso Ing. del conocimiento.

Objetivo: Determinar la función del Ing. del conocimiento.

Actor: Ingeniero del conocimiento.

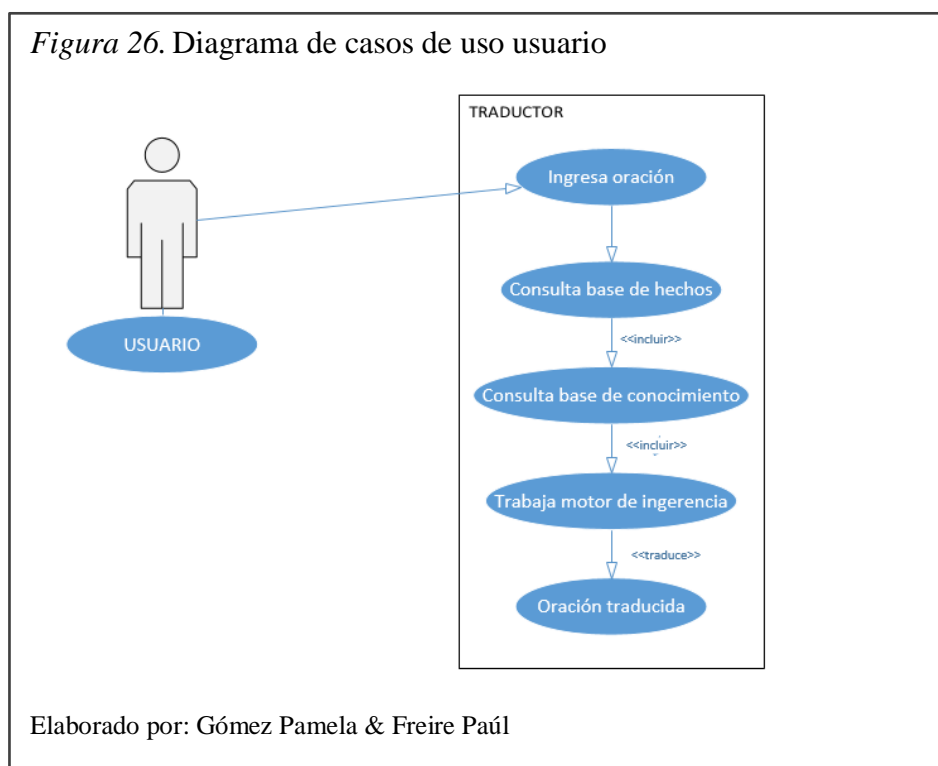
Pre-condición: Tener conocimientos básicos del idioma Kichwa.

Descripción:

El Ingeniero del conocimiento tiene la capacidad de ingresar hechos, reglas; poder interactuar en la base del conocimiento; para poder modificar el sistema experto.

Post-condición: Poder modificar los datos tomando en cuenta el conocimiento que el experto proporciona y lo traduce en lenguaje de programación.

2.3.3.3. Diagrama de casos de uso del usuario.



Nombre de caso de uso: Caso de uso usuario

Objetivo: Determinar la función del usuario

Actor: Usuario

Pre-condición: Tener estructurado el texto a traducir

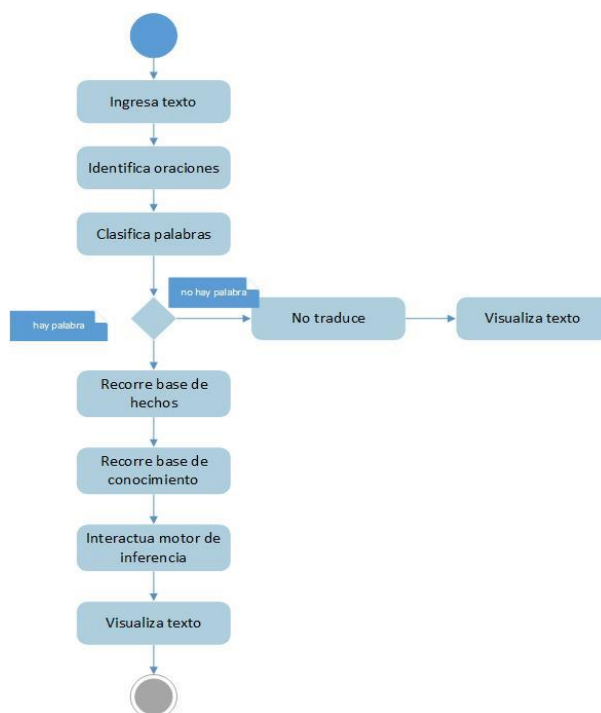
Descripción:

El usuario realiza el ingreso del texto a traducir, y espera tener la traducción más coherente y entendible.

Post-condición: El usuario puede hacer uso del sistema cuantas veces sea necesario.

2.3.4. Diagrama de flujo del sistema.

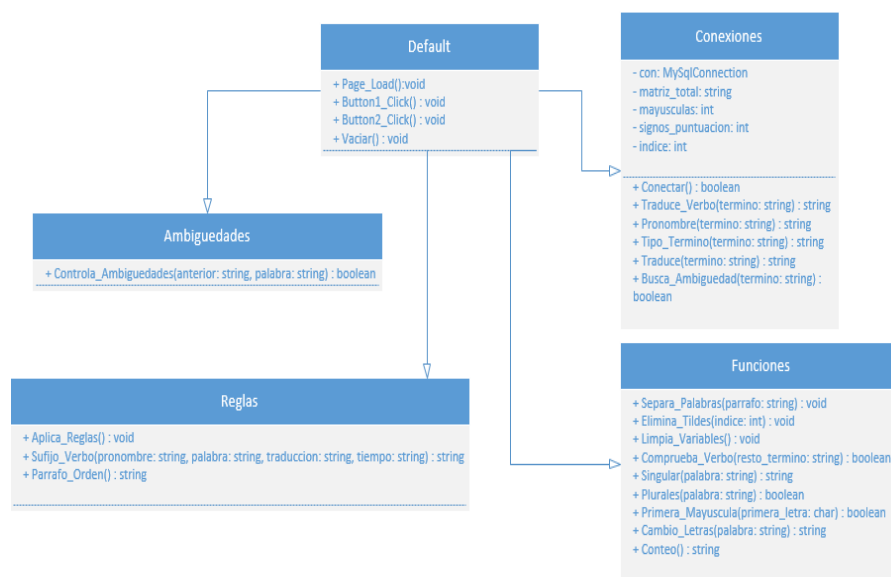
Figura 27. Diagrama de flujo del sistema



Elaborado por: Gómez Pamela & Freire Paúl

2.3.5. Diagrama de clases del sistema.

Figura 28. Diagrama de clases



Elaborado por: Gómez Pamela & Freire Paúl

2.4. Construcción

A continuación se detalla los principales componentes de la programación del sistema, y su construcción en base a clases, así:

- Clase conexiones.
- Clase funciones.
- Clase default
- Clase reglas de inferencia (sistema experto).
- Clase ambigüedades.
- Clase módulo de justificación (sistema experto).

Dichas clases contienen funciones que serán descritas mediante la presentación de ciertas partes relevantes de código fuente perteneciente al sistema, dichas partes permiten la explicación de cada clase. Cabe mencionar que el software del sistema de traducción es un solo paquete el cuál será detallado a continuación.

2.4.1. Clase conexiones.

Dentro de este módulo se definen las variables globales a trabajar en todo el sistema, es decir, la matriz que contendrá la traducción de todo el párrafo. También se especifica todas las conexiones que se realizarán con el gestor de base de datos, en este caso MySQL.

Figura 29. Clase conexiones

Conexiones
<ul style="list-style-type: none">- con: MySqlConnection- matriz_total: string- mayusculas: int- signos_puntuacion: int- indice: int
<ul style="list-style-type: none">+ Conectar() : boolean+ Traduce_Verbo(termino: string) : string+ Pronombre(termino: string) : string+ Tipo_Termino(termino: string) : string+ Traduce(termino: string) : string+ Busca_Ambigüedad(termino: string) : boolean

Elaborado por: Gómez Pamela & Freire Paúl

Clase: VariablesGlobales

Esta clase permite la creación de variables que gobernarán todo el sistema, además de las variables que controlan la conexión con la base de datos, así:

Figura 30. Clase variables globales

```
Public Class VariablesGlobales
    Public Shared con As MySqlConnection
    Public Shared matriz_total(3000, 5) As String
    Public Shared numeracion_reglas(100, 2) As String
    Public Shared mayusculas As String = "ABCDEFGHIJKLMNOPQRSTUVWXYZ"
    Public Shared signos_puntuacion As String = ",;:¡!¿?-()[\]{}"
End Class
```

Elaborado por: Gómez Pamela & Freire Paúl

Función: Conectar

Esta función contiene las credenciales para la conexión con la base de datos. Será llamada al iniciar el software y controla errores mediante la sentencia try catch, así:

Figura 31. Función conectar

```
Public Function Conectar() As Boolean
    Dim respuesta As Boolean = False
    Try
        Dim Servidor As String = "localhost"
        Dim Usuario As String = "root"
        Dim Pass As String = ""
        Dim BasedeDatos As String = "traductor"
        VariablesGlobales.con = New MySqlConnection()
        VariablesGlobales.con.ConnectionString = "server=" & Servidor & ";" _
            & "user id=" & Usuario & ";" _
            & "password=" & Pass & ";" _
            & "database=" & BasedeDatos & ""
        VariablesGlobales.con.Open()
        respuesta = True
        VariablesGlobales.con.Close()
    Catch ex As Exception
        VariablesGlobales.con.Dispose()
    End Try
    Return respuesta
End Function
```

Elaborado por: Gómez Pamela & Freire Paúl

Función: Traduce_Base_Verbo

Realiza una búsqueda en la tabla términos de una palabra que podría ser la base de un verbo en una oración; Devuelve como respuesta, en caso de encontrar una coincidencia, que corresponde a la traducción de dicho término, así:

Figura 32. Función traduce verbo

```
Public Function Traduce_Base_Verbo(termino As String) As String
    Dim respuesta As String = ""
    Dim table As New DataTable
    Dim sentencia As String
End Function
```

Elaborado por: Gómez Pamela & Freire Paúl

Función: Pronombre

Realiza una búsqueda del parámetro término, para encontrar una coincidencia para palabras que pueden ser pronombres personales. Devuelve un valor respuesta en caso de encontrar alguna traducción, así:

Figura 33. Función pronombre

```
Public Function Traduce_Pronombre(termino As String) As String
    Dim respuesta As String = ""
    Dim table As New DataTable
    Dim sentencia As String =
End Function
```

Elaborado por: Gómez Pamela & Freire Paúl

Función: Tipo_Termino

Realiza una búsqueda de un término en la que solamente devuelve el tipo de término al que corresponde, así:

Figura 34. Función tipo término

```
Public Function Tipo_Termino(termino As String) As String
    Dim respuesta As String = ""
    Dim table As New DataTable
    Dim sentencia As String =
End Function
```

Elaborado por: Gómez Pamela & Freire Paúl

Función: Traduce

Realiza una búsqueda de una palabra y devuelve su traducción directa desde la base de datos o base de hechos, así:

Figura 35. Función traduce

```
Public Function Traduce(termino As String) As String
    Dim respuesta As String = ""
    Dim table As New DataTable
    Dim sentencia As String = "SELECT traduccion FROM terminos WHERE
End Function
```

Elaborado por: Gómez Pamela & Freire Paúl

Función: Busca_Ambigüedad

Esta función realiza una búsqueda del término que entra como parámetro y devuelve un valor booleano que corresponde a que: si existen varias traducciones de un mismo término, si la respuesta es True, entonces entra al análisis de ambigüedades, así:

Figura 36. Función busca ambigüedad

```
Public Function Busca_Ambigüedad(termino As String) As Boolean
    Busca_Ambigüedad = False
    Dim table As New DataTable
    Dim sentencia As String = "SELECT traduccion FROM terminos WHERE
termino='" + LCase(Trim(CStr(termino))) + "'"
    End Function
```

Elaborado por: Gómez Pamela & Freire Paúl

2.4.2. Clase funciones.

Este módulo contiene las funciones para el procesamiento del texto ingresado, como la separación de palabras, eliminación de tildes, etc. Esto “depurará” la matriz para ingresar al proceso de traducción.

Figura 37. Clases funciones

Funciones
+ Separa_Palabras(parrafo: string) : void
+ Elimina_Tildes(indice: int) : void
+ Limpia_Variables() : void
+ Comprueba_Verbo(resto_termino: string) : boolean
+ Singular(palabra: string) : string
+ Plurales(palabra: string) : boolean
+ Primera_Mayuscula(primer_letra: char) : boolean
+ Cambio_Letras(palabra: string) : string
+ Conteo() : string

Elaborado por: Gómez Pamela & Freire Paúl

Función: Separa_Palabras

Esta función realiza el proceso de separación de palabras y las guarda en una matriz para su posterior traducción. El proceso toma en consideración los signos de puntuación punto, coma y signo de pregunta, así:

Figura 38. Función separa palabras

```
Public Sub Separa_Palabras(parrafo As String)
    Dim palabra As String = ""
End Sub
```

Elaborado por: Gómez Pamela & Freire Paúl

Función: Elimina_Tildes

Esta función elimina las tildes de cada una de las palabras que contengan dicho signo de puntuación. Esto es necesario para realizar la correcta traducción, así:

Figura 39. Función elimina tildes

```
Public Sub Elimina_Tildes(ind As Integer)
    Dim nuevo As String
End Sub
```

Elaborado por: Gómez Pamela & Freire Paúl

Función: Limpia_Variables

Para evitar datos erróneos en la traducción, se limpia o se vuelve a default las variables que se usan para el proceso, principalmente la matriz que contiene las palabras, así:

Figura 40. Función limpia variables

```
Public Sub Limpia_Variables()  
    For x As Integer = 0 To 3000  
        For y As Integer = 0 To 5  
            VariablesGlobales.matriz_total(x, y) = ""  
        Next y  
    Next x  
    VariablesGlobales.indice = 0  
End Sub
```

Elaborado por: Gómez Pamela & Freire Paúl

Función: Comprueba_Verbo

Función que devuelve un valor verdadero o falso según el caso, analiza si una palabra que está considerada como base de un verbo regular es definitivamente un verbo revisando su terminación, así:

Figura 41. Función comprueba verbo

```
Public Function Comprueba_Verbo(resto_termino As String) As Boolean  
    Comprueba_Verbo = False  
End Function
```

Elaborado por: Gómez Pamela & Freire Paúl

Función: Singular

En esta función realiza el reconocimiento de una palabra en singular, y busca en la base de datos su traducción, si la encuentra, devuelve su traducción, así:

Figura 42. Función singular

```
Public Function Singular(palabra As String) As String
    Singular = ""
    Dim letras(palabra.Length) As String
    Dim singular1 As String = ""
    Dim singular2 As String = ""
End Function
```

Elaborado por: Gómez Pamela & Freire Paúl

Función: Plurales

En esta función realiza el reconocimiento de una palabra en plural, así:

Figura 43. Función plurales

```
Public Function Plurales(palabra As String) As Boolean
    Plurales = False
    Dim letras(palabra.Length) As String
    Dim singular As String = ""
    Dim singular2 As String = ""
End Function
```

Elaborado por: Gómez Pamela & Freire Paúl

Función: Primera_Mayúscula

Determina si una palabra está en mayúsculas o no, así:

Figura 44. Función primera mayúscula

```
Public Function Primera_Mayuscula(primer_letra As Char) As Boolean
    Dim respuesta As Boolean = False
End Function
```

Elaborado por: Gómez Pamela & Freire Paúl

Función: Cambio_Letras

Esta función realiza el cambio: e por i, o por u y Q por K, en las palabras propias, así:

Figura 45. Función cambio de letras

```
Public Function Cambio_Letras(palabra As String) As String
    Dim cambiodeletras As String = ""
    Dim auxiliar As String = ""
    Dim Letra As Char
End Function
```

Elaborado por: Gómez Pamela & Freire Paúl

Función: Conteo

Realiza un conteo de palabras y oraciones presentes en el texto a traducir, así:

Figura 46. Función conteo

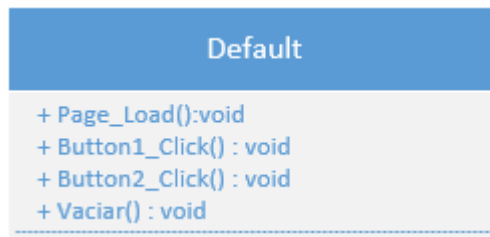
```
Public Function Conteo() As String
    Conteo = ""
    Dim contador As Integer = 0
End Function
```

Elaborado por: Gómez Pamela & Freire Paúl

2.4.3. Clase: default.

Es la clase principal que controla el procesamiento para traducción del texto. Contiene la función que ejecuta tanto la traducción directa, como el análisis del tipo de palabra que hubiere en el texto. Controla los verbos y muestra los resultados en la pantalla.

Figura 47. Clase default



Elaborado por: Gómez Pamela & Freire Paúl

Función: Button1_Click

Es la base misma del traductor, donde realiza todos los cambios, análisis de palabras, verbos, traducciones desde la base de datos, ordenamiento de las oraciones y salida del texto traducido obtenido como respuesta el análisis mediante reglas de inferencia, así:

Figura 48. Función button1

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles
Button1.Click
    Limpia_Variables()
    Dim letra_indice As Integer = 0
    Dim letras(100) As Char
    Dim nueva_palabra As String = ""
    Dim signo As Boolean = False
    Separa_Palabras(Trim(CStr(TextBox1.Text)))
    Elimina_Tildes(VariablesGlobales.indice)
    For x As Integer = 0 To VariablesGlobales.indice
        Dim palabra_evaluar As String = ""
        Dim traduccion_resp As String = ""
        Dim encontro As Boolean = False
        Dim primera_es_mayusculas As Boolean = False
        For letra_indice = 1 To (VariablesGlobales.matriz_total(x,
End Sub
```

Elaborado por: Gómez Pamela & Freire Paúl

Función: Button2_Click

Realiza el mismo análisis anterior, es decir, trabaja en base a reglas de inferencia, pero, además usa razonamiento aproximado para la detección de ambigüedades en el texto a traducir y realiza el tratamiento correspondiente para obtener como salida, una traducción mucho más coherente, así:

Figura 49. Función button2

```
Private Sub Button2_Click(sender As Object, e As EventArgs) Handles
Button2.Click
    Limpia_Variables()
    Dim letra_indice As Integer = 0
    Dim letras(100) As Char
    Dim nueva_palabra As String = ""
    Dim signo As Boolean = False
    Separa_Palabras(Trim(CStr(TextBox1.Text)))
End Sub
```

Elaborado por: Gómez Pamela & Freire Paúl

Función: Vaciar

Esta función solamente hace un vaciado de los cuadros de texto para nuevas traducciones.

2.4.4. Clase reglas de inferencia (sistema experto).

Módulo que especifica las reglas que se aplicarán a la matriz, luego de que haya pasado por el proceso de traducción. Estas reglas son predefinidas según el idioma y traducidas a lenguaje de programación. También realiza la verificación del tiempo en el que se trata cada oración.

Figura 50. Clase reglas de inferencia

Reglas
+ Aplica_Reglas() : void + Sufijo_Verbo(pronombre: string, palabra: string, traduccion: string, tiempo: string) : string + Parrafo_Orden() : string

Elaborado por: Gómez Pamela & Freire Paúl

Función: Aplica_Reglas

En esta función aplica las reglas generales de la traducción al idioma kichwa ecuatoriano, como los sufijos, artículos, signos de puntuación, etc, así:

Figura 51. Función aplica reglas

```
Public Sub Aplica_Reglas()  
    Dim pronombre As String = ""  
    Dim articulo As Boolean = False  
End Sub
```

Elaborado por: Gómez Pamela & Freire Paúl

Función: Sufijo_Verbo

Esta función analiza el tiempo en el que la oración está tratándose, haciendo referencia al pronombre usado, es decir, de quien se habla en la oración, y de las terminaciones de los verbos para así dar una respuesta aproximada del tiempo en el que se está hablando, así:

Figura 52. Función sufijo verbo

```
Public Function Sufijo_Verbo(pronombre As String, palabra As String,
traduccion As String, tiempo As String) As String
    Sufijo_Verbo = ""
    Dim letras(palabra.Length) As String
    Dim presente(6) As String
End Function
```

Elaborado por: Gómez Pamela & Freire Paúl

Función: Parrafo_Orden

Realiza la transformación de la matriz, para dar el orden en el que se mostrarán las oraciones basándose en las reglas del idioma Kichwa, así:

Figura 53. Función párrafo orden

```
Public Function Parrafo_Orden() As String
    Parrafo_Orden = ""
    Dim verbo As Integer = 0
    Dim m As Boolean = False
End Function
```

Elaborado por: Gómez Pamela & Freire Paúl

Función: Números

Realiza la traducción de números escritos en cifras a números cardinales, así:

Figura 54. Función números

```
Public Function Numeros(numero As String) As String
    Numeros = ""

    Dim num(13) As String
    Dim separa(6) As Integer
    Dim x As Integer
    Dim res As Integer
    Dim div As Integer = CInt(numero)
End Function
```

Elaborado por: Gómez Pamela & Freire Paúl

2.4.5. Clase ambigüedades.

En esta clase muy necesaria para la traducción completa, existe el módulo basado en razonamiento aproximado, que controla la mayoría de las ambigüedades léxicas detectadas en el texto traducido en el módulo basado en reglas de inferencia, para mejorar la traducción y hacerla más entendible al usuario.

Figura 55. Clase ambigüedades



Elaborado por: Gómez Pamela & Freire Paúl

Función: Controla_Ambigüedades

Mediante el uso de esta función, determinamos las palabras que contiene la ambigüedad léxica, para tratarla y dar una respuesta según sea el caso, así:

Figura 56. Función controla ambigüedades

```
Public Function Controla_Ambigüedades(anterior As String, palabra As String) As Boolean
    Controla_Ambigüedades = False
End Function
```

Elaborado por: Gómez Pamela & Freire Paúl

2.4.6. Clase módulo de justificación (sistema experto).

Esta clase permite visualizar toda la información requerida para el módulo de justificación, la cual permite conocer las reglas con su respectiva inferencia, proceso paso a paso, con el fin de justificar la traducción del sistema

Figura 57. Clase módulo de justificación

```
Public Class Justificacion
    Inherits System.Web.UI.Page

    Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Load
        ListBox1.Items.Clear()
        Numeración_Reglas()
    End Sub

    Private Sub Numeración_Reglas()
        Dim resultados(1000, 1) As String
    End Class
```

Elaborado por: Gómez Pamela & Freire Paúl

CAPÍTULO 3

PRUEBAS, RESULTADOS Y EVALUACIÓN

3.1. Pruebas

A continuación es necesario poner a prueba al sistema construido en el subcapítulo anterior, para lo cual se ha diseñado una tabla en donde se enuncia los requerimientos funcionales del sistema detallados anteriormente en el subcapítulo de diseño de la aplicación.

3.1.1. Pruebas contra requerimientos.

El objetivo de realizar las pruebas contra requerimientos es verificar si la aplicación los satisface. Para este subcapítulo fue necesario realizar las pruebas con usuarios y datos reales, y así obtener resultados lo más aproximados a la realidad.

3.1.1.1. *Requerimientos funcionales.*

Usuario del sistema

Tabla 11. *Pruebas contra requerimientos*

No.	Nombre de la prueba	Req. Func.	Procedimiento	Cumple	Observaciones
1	Indicar al usuario una interfaz amigable.	01	1. Acceder a la aplicación mediante un navegador web. 2. Explorar la interfaz de usuario	Sí	
2	Familiarizar al usuario con la interfaz.	01	1. Navegar a través de la interfaz 2. Conocer la utilidad del sistema	Sí	

continúa...

Tabla 12. *Pruebas contra requerimientos*

continuación...

No.	Nombre de la prueba	Req. Func.	Procedimiento	Cumple	Observaciones
3	Traducción de manera coherente y entendible.	02	1. Digitar texto en castellano a traducir. 2. Pulsar el botón con la etiqueta, traducir.	Sí	Se deberá tomar en cuenta signos de ortografía y puntuación al momento del ingreso de texto.
4	El usuario confía en la coherencia del resultado de la traducción.	02	1. Comprender el texto traducido	Sí	Para este proceso de indica al usuario el mismo texto traducido por el sistema, traducido por un experto en el idioma.
5	Traducción con detección de ambigüedades léxicas.	03	1. Digitar texto en castellano a traducir. 2. Pulsar el botón con la etiqueta, traducir. 3. Visualizar palabras que contengan ambigüedades léxicas.	Sí	
6	Traducción sugerida.	04	1. Pulsar el botón con la etiqueta, traducción sugerida	Sí	El sistema mejoró la traducción aun presentando ambigüedades léxicas mínimas.
7	Traducción en presente, pasado y futuro.	05	1. Ingresar texto en idioma castellano en los tiempos presente, pasado y futuro indistintamente al sistema. 2. Pulsar el botón con la etiqueta, traducir.	Sí	

Elaborado por: Gómez Pamela & Freire Paúl

3.2. Resultados

Después de concluir con el desarrollo de la aplicación, se dispone de un sistema experto traductor de los idiomas: Castellano a Kichwa ecuatoriano.

A continuación se expone el manual de usuario que muestra el funcionamiento del sistema.

3.2.1. Acceso al sistema.

Para acceder al sistema es necesario disponer de un navegador web, ya sea Internet Explorer (7 o superior) o Mozilla Firefox. Se ingresa la dirección IP del servidor en el cual se encuentra el sistema. Para el acceso se utilizará la dirección IP del ordenador local (localhost), así:

(<http://localhost:59880/Prueba.aspx>)

Figura 58. Interfaz de usuario



Elaborado por: Gómez Pamela & Freire Paúl

Ya desplegada la página de inicio se podrá visualizar la interfaz del usuario del sistema, donde se dispone de títulos, cuadros de texto y botones, de los cuales se explicará seguida y detalladamente paso a paso su funcionalidad. Para mayor facilidad de explicación, se ha dividido la interfaz en dos módulos.

Dicha interfaz cuenta con todo lo necesario para cumplir con los objetivos de funcionalidad del sistema y no fue necesario el diseño de más páginas HTML.

3.2.1.1. Módulo 1.

En el presente módulo se visualiza el respectivo cuadro de texto donde necesariamente se debe ingresar texto en idioma castellano, también se dispone de un botón con la etiqueta “Traducir“, el cual después de ser accionado mediante un clic, traduce mencionado texto al idioma Kichwa ecuatoriano.

Es necesario recalcar que el texto a ingresar debe apegarse en lo posible a las reglas de escritura del idioma castellano, es decir tomar muy en cuenta signos de ortografía y puntuación, en orden de obtener una buena traducción.

También es importante decir que al término de una oración es necesario colocar un punto o coma según el caso, dado que esto se apega a las reglas de escritura del idioma castellano y también facilita el proceso del sistema.

Figura 59. Módulo 1



Elaborado por: Gómez Pamela & Freire Paúl

3.2.1.2. Módulo 2.

Basándose en reglas de inferencia, este módulo presenta la traducción del texto ingresado anteriormente en el módulo uno. Dada la necesidad de presentar una opción para tratar las ambigüedades léxicas en este módulo, se subdivide en dos casos, así.

- Caso 1: Texto sin ambigüedades léxicas.

Además del texto original, en este fragmento se visualiza el título “Traducción basada en reglas de inferencia”, conjuntamente con el texto traducido al idioma Kichwa ecuatoriano.



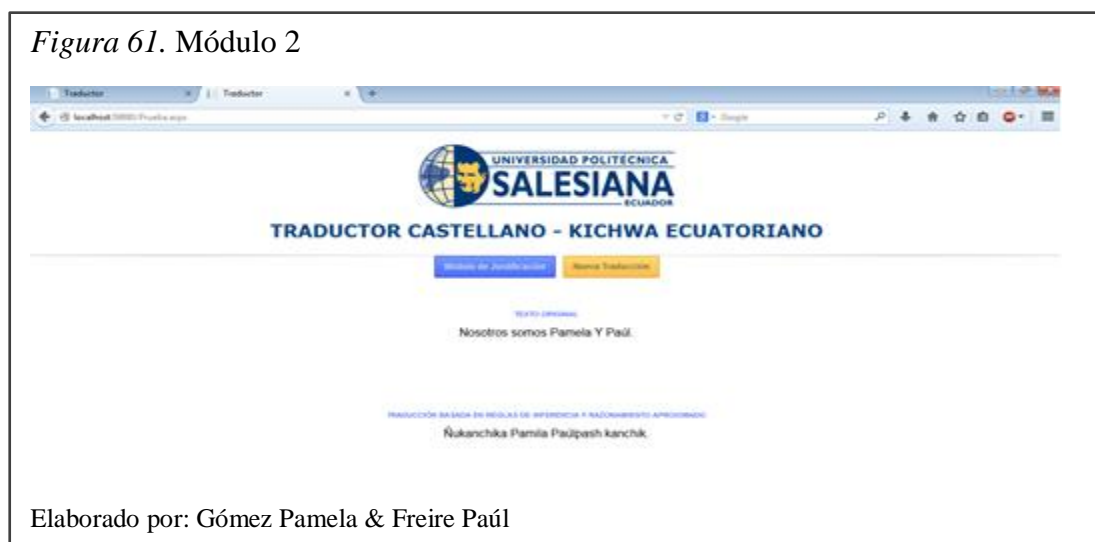
También se presentan los botones con las etiquetas: “Traducción Sugerida”, “Módulo de Justificación” y “Nueva traducción”, respectivamente, así:

- Traducción Sugerida.

Se visualiza a partir del botón con la etiqueta “Traducción Sugerida”, al dar clic sobre el mismo aparece el texto “Traducción basada en reglas de inferencia y razonamiento aproximado”, conjuntamente con el texto traducido al idioma Kichwa ecuatoriano.

Mediante este fragmento se visualiza la traducción sugerida por el sistema, la cual se basa en el conocimiento del mismo, dicho conocimiento lo obtiene a partir de las sugerencias ingresadas repetidamente por el usuario en el módulo de selección, conjuntamente con las reglas de inferencia. El sistema aprenderá y evaluará las sugerencias en base a la experiencia del usuario, y realizará un balance probabilístico para presentar un resultado coherente, automáticamente sin necesidad de volver a preguntar al usuario.

Figura 61. Módulo 2



- Módulo de Justificación.

Al presionar el botón con la etiqueta “Módulo de Justificación”, se visualizará mencionado módulo, llamado también interfaz explicativa, donde el sistema mostrará el número de palabras y oraciones encontradas, también demostrará: el tipo de palabras, número de reglas, nombre y las reglas utilizadas para realizar la traducción y obtención de un resultado coherente.

Adicionalmente dentro del módulo de justificación, se visualiza el botón regresar que permite realizar una nueva traducción.

Figura 62. Módulo 2



Elaborado por: Gómez Pamela & Freire Paúl

- Nueva Traducción.

Al presionar el botón con la etiqueta “Nueva traducción”, el sistema regresará al módulo uno, para empezar nuevamente con otro proceso de traducción.

Figura 63. Módulo 2



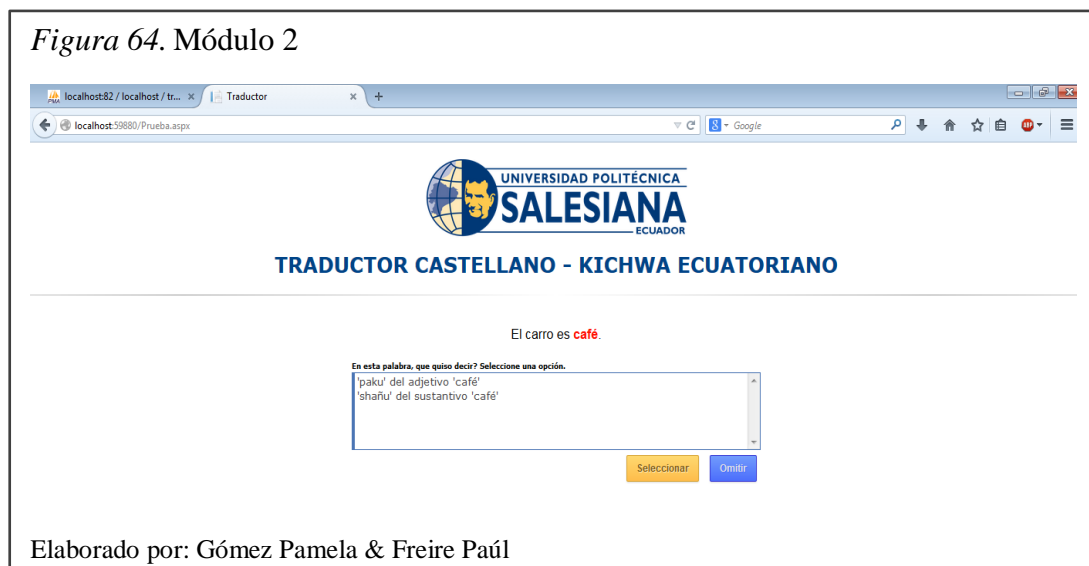
Elaborado por: Gómez Pamela & Freire Paúl

- Caso 2: Texto con ambigüedades léxicas.

Aquí se visualiza el texto original, el cual contiene la palabra que presenta ambigüedad léxica resaltada de color rojo, también se despliega el título “En esta palabra, quiso

decir? Seleccione una opción”, conjuntamente con el cuadro de texto donde se presenta la palabra con ambigüedad léxica resaltada anteriormente, se da la alternativa de seleccionar la opción más adecuada en orden de complacer la necesidad de traducción del usuario, y a su vez se facilita el aprendizaje del sistema a base de la experiencia del usuario.

Figura 64. Módulo 2



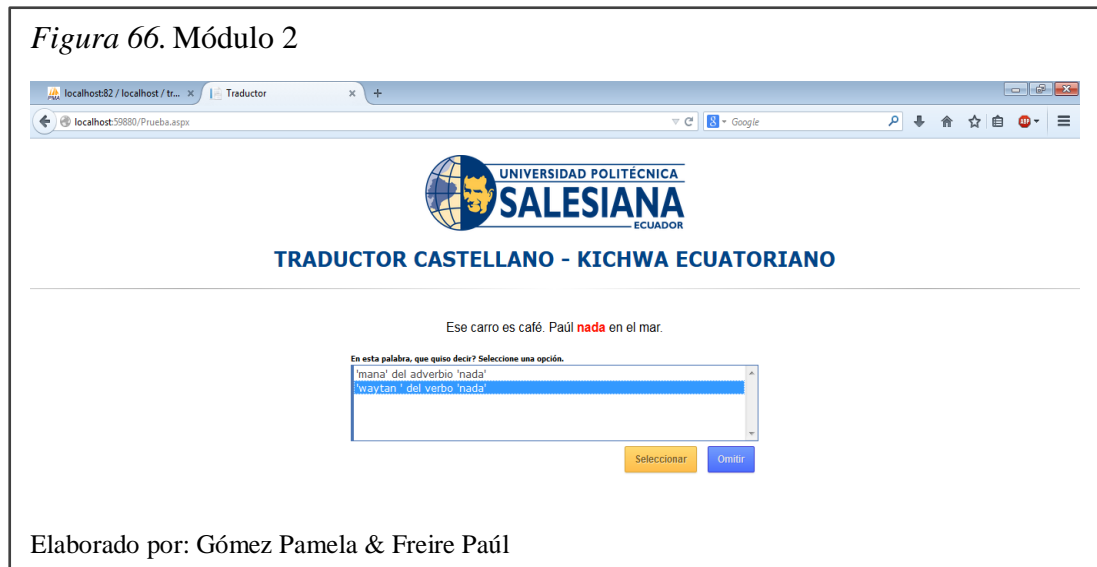
Para realizar la validación de dicha selección es necesario pulsar el botón con la etiqueta “Seleccionar”, caso contrario al presionar el botón con la etiqueta “Omitir” el sistema presentará como resultado la traducción más acertada posible en base a su conocimiento.

Figura 65. Módulo 2



En el caso de presentarse varias oraciones, el sistema mostrará ordenadamente una a una las palabras que presentan ambigüedad léxica, resaltándolas oración por oración, permitiendo al usuario seleccionar e interactuar con la interfaz de manera fácil, hasta que el sistema aprenda, para después realizar la traducción automáticamente.

Figura 66. Módulo 2



Una vez terminada la selección de opciones, y después de pulsar el botón con la etiqueta “Seleccionar”, el sistema desplegará el resultado de traducción que incluye las sugerencias ingresadas por el usuario.

Figura 67. Módulo 2



También se presentan nuevamente los botones con las etiquetas: “Traducción Sugerida”, “Módulo de Justificación” y “Nueva traducción”, respectivamente, los cuales realizan las mismas acciones descritas anteriormente.

3.3. Evaluación

Para diseñar la evaluación, se tuvo que realizar un análisis y estudio básico de los parámetros que determinan la calidad de traducción de un texto, con la ayuda del experto en el idioma, resumimos los siguientes:

- Morfología
- Sintaxis.
- Semántica.
- Pragmática.

Dándose la posibilidad que las personas que van al evaluar el texto traducido por el sistema, no comprendan de manera clara y concisa los parámetros enunciados anteriormente se vio la necesidad de traducirlos a lenguaje natural, así:

- Morfología → Comprensión de la traducción.
- Sintaxis → Gramática de la traducción.
- Semántica → Coherencia de la traducción.
- Pragmática → Confiabilidad de la traducción.

Estos conceptos se encuentran detalladamente en este trabajo escrito.

3.2.1. Modelo de la evaluación.

Tomando en cuenta lo mencionado en el punto anterior, y todo lo requerido para la buena comprensión y desarrollo del evaluador, se diseñó el siguiente modelo de evaluación, resaltando que:

- Traducción 1: es la traducción realizada por el experto en el idioma.
- Traducción 2: es la traducción de un sistema encontrado en la web.

- Traducción 3: es la traducción de nuestro sistema basado en reglas de inferencia.
- Traducción 4: es la traducción de nuestro sistema usando los métodos basados en reglas de inferencia y razonamiento aproximado.

3.2.2. Resultados de la evaluación.

La evaluación fue realizada a estudiantes del departamento de idiomas (Kichwa) de la Universidad Central del Ecuador, después de la cual se obtuvieron los siguientes resultados:

- **Traducción 1 (Experto)**

Siendo 19.1 puntos igual al 100%, la traducción realizada por el experto obtuvo la totalidad de 19.1 puntos, es decir el 100% en la calidad de traducción.

- **Traducción 2 (Kichwas)**

Siendo 19.1 puntos igual al 100%, la traducción realizada por el sistema Kichwas obtuvo la totalidad de 2.1 puntos, es decir el 10.99% en la calidad de traducción.

- **Traducción 3 (Sistema basado en reglas)**

Siendo 19.1 puntos igual al 100%, la traducción realizada por el sistema basado en reglas obtuvo la totalidad de 7.5 puntos, es decir el 39.26% en la calidad de traducción.

- **Traducción 4 (Sistema basado en reglas y razonamiento aproximado)**

Siendo 19.1 puntos igual al 100%, la traducción realizada por el sistema basado en reglas y razonamiento aproximado obtuvo la totalidad de 8.2 puntos, es decir el 42.93% en la calidad de traducción.

Dados estos resultados podemos mencionar que la calidad de traducción del sistema basado en reglas de inferencia y razonamiento aproximado mejoró en un 3.67%, con respecto al sistema que únicamente trabaja con reglas de inferencia.

Cabe resaltar que la complejidad de la traducción para la evaluación es básica y no presenta mayor reto tanto para los evaluadores como para el sistema.

CONCLUSIONES

- Después de aplicar el módulo del sistema experto traductor basado solamente en reglas de inferencia a un texto en idioma Castellano, se visualizó que la traducción fue coherente y aceptable, pero contenía cierto porcentaje de ambigüedades léxicas, las cuales fueron resueltas con la ayuda del usuario al ingresar sugerencias en base a su experiencia con respecto al manejo de palabras similares. Al anexar el conocimiento adquirido por parte del usuario en mencionado módulo al método de razonamiento aproximado del programa, se logró el aprendizaje del sistema, el cual en base a dicho aprendizaje, resolvió automáticamente las ambigüedades léxicas y mejoró la coherencia de la traducción, pero siempre se obtuvo un menor porcentaje de error, por lo cual se concluye también que se trata de un sistema informático y no de un experto humano, ese margen de error es el que diferencia hasta ahora a la inteligencia y razonamiento humano con el procesamiento de la máquina.
- Mediante la traducción automática basada en reglas de inferencia y razonamiento aproximado se concluyó que el procesamiento de lenguaje natural es una de las mejores técnicas a utilizar en el campo de la traducción ya que resuelve gran parte de problemas relacionados con las palabras y sus posibles distintas traducciones dependiendo del texto contiguo usando probabilidad, dando así coherencia a los resultados.
- Al utilizar la metodología orientada a objetos en una de sus fases como es el diseño orientado a objetos, logramos modelar el sistema a una interacción lógica y sencilla entre sus componentes, en este caso llamados objetos, dando solución a el manejo de gran cantidad de información suministrada por el experto y también de código generado por los desarrolladores, también se logró interactuar con varios componentes del sistema experto ordenadamente y obedeciendo a realizar una aceptable ingeniería del software utilizando el lenguaje unificado de modelado que facilitó la visualización, construcción y documentación del mismo.

- Para lograr un buen diseño del sistema fue necesario el gran aporte del experto, se contó con la suerte de encontrar una persona con gran capacidad de enseñanza de calidad y colaboración, convirtiéndose así en el componente principal del sistema experto, su información fue esencial e indispensable en este proyecto.
- Luego de la investigación y estudio del idioma Kichwa se concluyó que no existe un organismo que lo reglamente como lo es la Real Academia Española para la lengua Castellana cuya principal función es fijar las voces y vocablos de la dicha lengua.

En el Ecuador organizaciones como la CONAIE o el Departamento de Interculturalidad Bilingüe perteneciente al ministerio de educación, no han logrado reglamentar el idioma Kichwa de manera general, ya que las personas Kichwa hablantes poseen diferentes clases de ideologías con respecto al habla y escritura del idioma Kichwa, razón por la cual no se ha podido unificar en un solo documento una forma general de manejar dicho idioma como ya lo ha hecho la antes mencionada Real Academia Española para el idioma Castellano.

Por esta razón se encontrarán varios textos del idioma Kichwa ecuatoriano con algunas diferencias, una de las más principales es el manejo de la c en vez de k, y entre otras la manera de hablar y escribir el idioma. Actualmente se pretende rescatar y unificar el idioma nativo de Ecuador, Perú y Bolivia mediante la interacción de estos países pero, no se ha llegado a algún acuerdo oficial.

RECOMENDACIONES

- Para el mejor funcionamiento del sistema se recomienda alimentar la base de hechos, ya que esta contiene un diccionario básico del idioma Kichwa ecuatoriano de aproximadamente cuatro mil palabras. Tomar en cuenta que la lengua nativa es muy extensa y se puede encontrar variedad de palabras que no constan en dicho diccionario, en los pueblos indígenas aun existentes en nuestro país.
- El sistema es uno de los primeros traductores de Castellano a Kichwa ecuatoriano basado en un sistema experto, razón por la cual se ha tomado los métodos más frecuentes para traducción como lo son: traducción por medio de reglas de inferencia y traducción por razonamiento aproximado, por lo que es recomendable continuar con el estudio del tema y ampliarlo de manera que pueda realizar una traducción inteligente similar a la de los traductores existentes en la web para el idioma inglés, usando varios otros métodos existentes utilizados en traducción, cabe resaltar que el sistema queda abierto a cualquier mejora en beneficio de la investigación.
- La mejor opción para obtener una respuesta coherente en traducción por parte del sistema es ingresando texto coherente y acogiéndose a las normas generales del idioma Castellano como son: el uso de mayúsculas en los nombres propios, el correcto uso de los signos de puntuación, evitar en lo posible las faltas de ortografía, etc.
- Para la seguridad del sistema se recomienda la adición de un módulo de autenticación de ingreso de usuario con su respectiva contraseña, esto en caso de que se haga público el código fuente, ya sea para su mejora o manipulación directa por parte de personas autorizadas, este sistema podría ser de gran ayuda en la web, ya que contribuiría a la propagación del uso del idioma Kichwa ecuatoriano.

LISTA DE REFERENCIAS

- Abaitua, J. (2001). Introducción a la Traducción Automática. Universidad de Deusto: Grupo Deli.
- Acuña, E. (2010). Conceptos Básicos de Probabilidad. Puerto Rico: UPRM.
- Arrúa, L., & Meza Fernández, E. (2003). Inteligencia Artificial Sistemas Expertos. Buenos Aires: exa.unne.edu.ar.
- Berzal, F. (2010). Sistemas Experto . Granada: DECSAI Departamento de Ciencias de la Computación e IA.
- cala.unex.es. (2012). Hiperenciclopédia de divulgación del saber. Vol. 6, 4.
- Castillo, E., Gutiérrez, J. M., & Hadi, A. S. (2000). Sistemas Expertos y Modelos de Redes Probabilísticas. Santander: Universidad de Cantabria.
- Catarina. (2010). Lógica Difusa. México: UDLAP.
- Díez, F. J. (2005). Introducción al Razonamiento Aproximado. Madrid: UNED.
- disi.unal.edu.co. (s.f.). Inteligencia Artificial.
- ecured.cu. (2012). Procesamiento del Lenguaje Natural.
- Edward, F. (1965). Creador del primer sistema experto.
- esi.ue.es, j. (2013). Procesamiento de Lenguaje Natural.
- Galeon. (s.f.). La Robótica del Mundo.
- Giarratano, J., & Gary, R. (2001). Sistemas Expertos principios y programación. México: Thomson Editores, S.A. de C.V.
- Kendall, K., & Kendall, J. (2009). Análisis y diseño de sistemas. México: Pearson Educación.
- Mafu, m. (2013). Diagramas UML.
- Paza, G. M. (2011). KICHWA KAMU. Quito.
- Sosa, E. (1997). Procesamiento de Lenguaje Natural . Barcelona, España: El profesional de la información.
- Sucar, L. E. (2010). Redes Bayesianas. Puebla: INAOE.
- Systran. (2014). Qué es la traducción automática. www.systranet.com.
- www.docentes.unal.edu.co. (2011). Sistemas Expertos Automatización. Bogotá: besusernag.

ANEXOS

Evaluaciones

A continuación adjuntamos las evaluaciones realizadas a los estudiantes de idiomas de la Universidad Central del Ecuador, alumnos del docente tutor experto Dr. Rommel Sánchez.

Anexo 1. *Evaluación del traductor Castellano – Kichwa (modelo 1)*

Bajo su conocimiento; favor evalúe tomando en cuenta los siguientes parámetros:

Muy bueno 5; Bueno 4; Regular 3; Malo 2; Insuficiente 1.

Yo soy profesor de idioma kichwa, mi nombre es Rommel Sánchez, yo enseñé kichwa en la Universidad Central, mi familia y yo somos de Otavalo, mi esposa y yo vivimos 20 años en Quito, el idioma kichwa es antiguo.

<p>Traducción 1:</p> <p>Ñukaka kichwa yachachik runami kani, ñukapak shutika Rummil Sanchezmi, ñukaka Cintral Univirsidadpi kichwata yachachirkani, ñuka aylluka ñukapash Utavalumanda kanchik, ñukapak warmika ñukapash ishki chunka watata Kitupi kawsanchik, shimika mawka kichwa kan.</p>	<p>Traducción 2:</p> <p>ñuka kani yachachik pak ref kichwa ñuka shuti kan Rommel Sanchez ñuka enseñe kichwa pi sumak-yachana- wasi Central ñuka ayllu pash ñuka somos pakkuna Otavalo ñuka harmi pash ñuka vivimos 20 watakuna pi Quito.</p>	<p>Traducción 3:</p> <p>Ñukaka yachachik shimipak kichwa kani, Ñukapak shuti Rummil Sánchez kan, Ñukaka kichwata Hamutaywasipi Cintral yachachirkani, Ñukapak ayllu ñukakapash Utavalumanda kanchik, Ñukapak warmi ñukakapash ishkay chunka wata Kitupi kawsani, Shimika kichwa ñawpa kan.</p>	<p>Traducción 4:</p> <p>Ñukaka yachachik shimimanda kichwa kani, Ñukapak shuti Rummil Sánchez kan, Ñukaka kichwata Hamutaywasipi Cintral yachachirkani, Ñukapak ayllu ñukakapash Utavalumanda kanchik, Ñukapak warmi ñukakapash ishkay chunka wata Kitupi kawsani, Shimika kichwa ñawpa kan.</p>
--	---	---	---

Parámetros	Traducción 1	Traducción 2	Traducción 3	Traducción 4
Comprensión de la traducción				
Gramática de la traducción				
Coherencia de la traducción				
Confiabilidad en la traducción				

Anexo 2. Evaluación del traductor Castellano – Kichwa (modelo 2)

Bajo su conocimiento; favor evalúe tomando en cuenta los siguientes parámetros:

Muy bueno 5; Bueno 4; Regular 3; Malo 2; Insuficiente 1.

Pedro tiene 10 años. Juan tiene 8 años. Ellos van hacia casa de su madre. Su madre es Josefina. Ella es profesora de Historia en Latacunga. Ellos viven en Pujilí.

<p>Traducción 1:</p> <p>Pidruka chunka watata charin. Juanka pusak watata charin. Paykunaka wasiman paypak mama rinkuna. Paypak mama Jusifina kan. Payka Historiapak yachachik Latacungapi kan. Paykunaka Pujilipi kawsankuna.</p>	<p>Traducción 2:</p> <p>Pedro tiene 10 años. Juan tiene 8 años. paykuna. van man wasi pak pron mama. pron mama kan Josefina. pron kan yachachik pak Historia pi Latacunga. paykuna. viven pi Pujilí.</p>	<p>Traducción 3:</p> <p>Pidru chunka wata charin. Juan pusak wata charin. Paykunaka wasiman ta paypakpak mama rinkuna. Paypak mama Jusifina kan. Payka yachachik Histuriapak Latacungapi kan. Paykunaka Pujilípita kawsankuna.</p>	<p>Traducción 4:</p> <p>Pidru chunka wata charin. Juan pusak wata charin. Paykunaka wasiman ta paypakmanda mama rinkuna. Paypak mama Jusifina kan. Payka yachachik Histuriamanda Latacungapi kan. Paykunaka Pujilípita kawsankuna.</p>
---	---	---	---

Parámetros	Traducción 1	Traducción 2	Traducción 3	Traducción 4
Comprensión de la traducción				
Gramática de la traducción				
Coherencia de la traducción				
Confiabilidad en la traducción				

Anexo 3. Evaluación del traductor Castellano – Kichwa (modelo 3)

Bajo su conocimiento; favor evalúe tomando en cuenta los siguientes parámetros:

Muy bueno 5; Bueno 4; Regular 3; Malo 2; Insuficiente 1.

Quito es la capital del Ecuador también la carita de dios. Esta ciudad tiene más de 1000000 de personas. Su altitud es 2800 msnm. Quito tiene el centro histórico más grande y preservado de América.

<p>Traducción 1:</p> <p>Kituka Ecuadorpak capital uyaka achillik- yayapakpash kan. Kay hatunllakta ashtawanta junopak runakunapakta charin. Paypak altitud ishkay waranka pusak patsak msnm kan. Kitu historico chawpimi ashtawanta hatun preservadopash Americapak charin.</p>	<p>Traducción 2:</p> <p>Quito kan ta capital pak Ecuador pash ta carita pak achillik-tayta. kay llakta tiene ashtawan pak 1000000 pak personas. pron. Quito tiene ta chawpi histórico ashtawan jatun pash preservado pak América.</p>	<p>Traducción 3:</p> <p>Kitu capital mandakuna Icuadur caritapash achillik - yayapak kan. Kay hatunllakta ashtawanta junu warankapak runakunapak charin. Paypak altitud ishkay waranka pusak patsak msnm kan. Kitu chawpita histórico ashtawan hatun preservadopash Pantinsayamanda charin.</p>	<p>Traducción 4:</p> <p>Kitu capital mandakuna Icuadur caritapash achillik - yayamanda kan. Kay hatunllakta ashtawanta junu warankamanda runakunamanda charin. Paypak altitud ishkay waranka pusak patsak msnm kan. Kitu chawpita histórico ashtawan hatun preservadopash Pantinsayamanda charin.</p>
--	--	--	--

Parámetros	Traducción 1	Traducción 2	Traducción 3	Traducción 4
Comprensión de la traducción				
Gramática de la traducción				
Coherencia de la traducción				
Confiabilidad en la traducción				

Anexo 4. Evaluación del traductor Castellano – Kichwa (modelo 4)

Bajo su conocimiento; favor evalúe tomando en cuenta los siguientes parámetros:

Muy bueno 5; Bueno 4; Regular 3; Malo 2; Insuficiente 1.

Había una vez una adorable niña que era querida por todo aquél que la conociera, pero sobre todo por su abuelita, y no quedaba nada que no le hubiera dado a la niña.

<p>Traducción 1:</p> <p>Ñukaka kichwa yachachik runami kani, ñukapak shutika Rummil Sanchezmi, ñukaka Cintral Univirsidadpi kichwata yachachirkani, ñuka aylluka ñukapash Utavalumanda kanchik, ñukapak warmika ñukapash ishki chunka watata Kitupi kawsanchik, shimika mawka kichwa kan.</p>	<p>Traducción 2:</p> <p>Había shuk vez shuk adorable wawa iwka era kuyashka rayku tukuy aquél iwka ta conociera, conj jawapi tukuy rayku pron abuelita pash mana quedaba imas iwka mana le hubiera dado ta ta niña.</p>	<p>Traducción 3:</p> <p>shuk rikunkuna shuk adorable pichiw que era querida por tukuy aquél que conocieraka, Shinallatak tukuyta por paypak abuelita puchu, Manapash quedaba mana que mana hubierata dado pichiwman killkakati.</p>	<p>Traducción 4:</p> <p>shukkuna rikunkuna shukkuna adorable pichiw que era querida por tukuy aquél que conocieraka, Shinallatak tukuyta por paypak abuelita puchu, Manapash quedaba mana que mana hubierata dado pichiwman killkakati.</p>
--	--	--	--

Parámetros	Traducción 1	Traducción 2	Traducción 3	Traducción 4
Comprensión de la traducción				
Gramática de la traducción				
Coherencia de la traducción				
Confiabilidad en la traducción				

EVALUACIÓN DEL TRADUCTOR CASTELLANO – KICHWA

Bajo su conocimiento; favor evalúe tomando en cuenta los siguientes parámetros:

Muy bueno 5; Bueno 4; Regular 3; Malo 2; Insuficiente 1.

Salve oh Patria mil veces oh Patria, gloria a ti, ya tu pecho rebosa, gozo y paz, y tu frente radiosa, mas que el sol contemplamos lucir.

<p>Traducción 1:</p> <p>Sumak llakta kantami napanchik. Kuyaywan kuyaywan kanka tukuy shunkuwan kushikuy. Ñuka kuyay intita yallinmi, waranka warankata kushikuy, kanka sumak allpami kanki.</p>	<p>Traducción 2:</p> <p>Salve oh llakta waranka veces oh llaktakuna gloria kanta ña kanpak pecho rebosa, gozo pash paz pash kanpak mayan radiosa shinalli iwka ta inti contemplamos lucir.</p>	<p>Traducción 3:</p> <p>Salvi chukuna Mamallakta waranka veces chukuna Mamallakta, Gloria timan, Ña kanka kasku rebosa, Gozo pazpash, Kankapash tirku radiosa, Mas que intika contemplamos lucir.</p>	<p>Traducción 4:</p> <p>Salvi chukuna Mamallakta waranka veces chukuna Mamallakta, Gloria timan, Ña kanka kasku rebosa, Gozo pazpash, Kankapash tirku radiosa, Mas que intika contemplamos lucir.</p>
---	---	--	--

Parámetros	Traducción 1	Traducción 2	Traducción 3	Traducción 4
Comprensión de la traducción				
Gramática de la traducción				
Coherencia de la traducción				
Confiabilidad en la traducción				